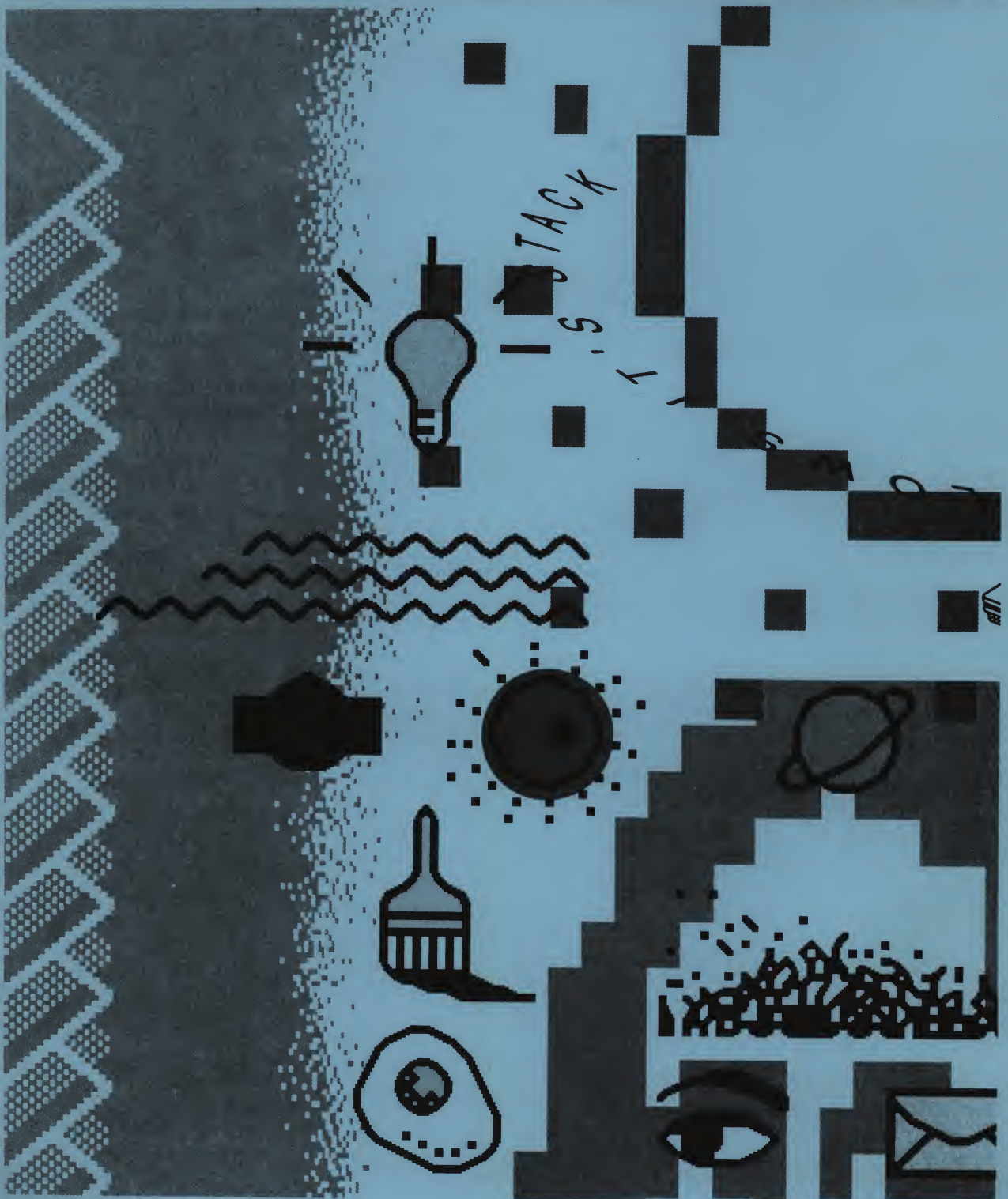


Augustus 1988 nr.37

# Quasar





# QUASAR

## INHOUD QUASAR 37

Kolofon .....	762
Van de Redactie .....	763
Lightning .....	764
Eprom programmer .....	770
Databanknieuws .....	773
Een klein kopje toe .....	773
Succes met CP/M .....	774
Qtalker voor zelfbouw .....	775
Cursus Machinetaal .....	778
OMNI Reader .....	779
Printer drivers .....	780
HARDWARE .....	782
Tandata Qconnect .....	784
Internationale QL-bijeenkomst .	785

## SLUITINGDATA KOPIJ

10 september  
8 oktober  
7 november

## KOLOFON

Stichting SIN QL AIR  
Rotterdam  
giro: 4597345

**ADMINISTRATIE**  
sekretariaat  
**PENNINGMEESTER**  
Nabestellen oude  
nummers.

**BOB VISSER**  
Scheepmakerskade 30  
3011 VX Rotterdam  
Tel. 010 - 414.3554

### VOORZITTER

**RON DEN BREEMS**  
Kroonstadeef 27  
3067 RT Rotterdam  
Tel. 010 - 455.1234

**REDAKTIE**, layout en  
samenstelling Quasar.

**GERARD VAN  
ROOIJEN**  
Gruttostraat 15  
3435 DJ Nieuwegein  
Tel. 03402 - 33027

**DATABANK**  
Tel. 03404 - 22533  
Sysop:  
**MICHEL & WILLEM  
SPANJER**  
Hortensialaan 11  
3702 VD Zeist  
Tel. 03404-20581  
**SVP aléén tussen**  
19.00 en 22.00 uur

**HARDWARE**  
Reparaties en  
onderdelen

**MICHEL & WILLEM  
SPANJER**  
Hortensialaan 11  
3702 VD Zeist  
Tel. 03404 - 20581  
**SVP alléén tussen**  
19.00 en 22.00 uur

**VRAGEN OVER:**  
Superbasic, Pascal,  
Machinetaal, Quill,  
Archive, Abacus,  
Hardware

**KEES V.D. WAL**  
Kwekerijstraat 22  
2613 VE Delft  
Tel. 015 - 140367  
**SVP alléén tussen**  
20.30 en 23.30 uur

**VRAGEN OVER:**  
Machinetaal en Hard-  
ware

**ARD JONKER**  
Tel. 020 - 230795





Zo de vakantie zit er weer op en werd het weer hoog tijd de nieuwe Quasar in elkaar te draaien.

Dit nummer hebben wij niet te klagen over het aanbod van hardware projecten, zoals de Eprommer van Jan Bredenbeek, het alternatief voor de Qtalker van Leo Moll, en Peter Venema met diverse uitbreidingen. Blijf het volgen.

De aanbieding van artikelen over het gebruik van Archive blijft wat achter dus mensen indien jullie een leuke toepassing in Archive en Abacus hebben geschreven laat het ons weten.

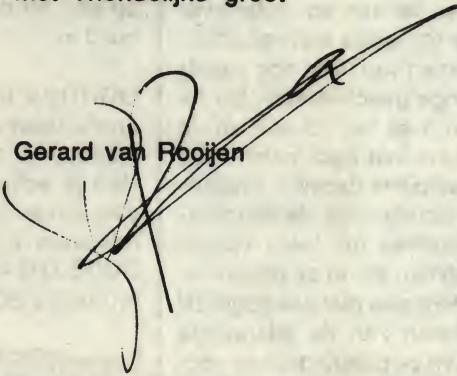
Door Oleg Holl en Arnold Spreekmeester worden diverse bijeenkomsten voor sinclair gebruikers georganiseerd in Blaricum. Ook de Europese QL dag op 23 oktober in België staat vast.  
(zie elders in de Quasar).

Wij blijven uitzien naar jullie inzenden over het QL-gebeuren.

Uw inzendingen gaarne als Quill-doc of als ASCII-file op cartridge of 5 1/4 inch floppy. Ook een uitgeprinte versie bij leveren is voor ons erg handig.

Met vriendelijke groet

Gerard van Rooijen

A large, stylized handwritten signature in black ink, written over the printed name 'Gerard van Rooijen'.



# L I G H T N I N G

Mij is gevraagd om een stukje over LIGHTNING, de tekst en graphics-versneller van Digital Precision te schrijven. Nu ik al enige tijd LIGHTNING in huis heb, ben ik voldoende vertrouwd met de meeste "ins" en "outs" dat ik er wat overkwijt kan. In dit stukje zal het volgende aan de orde komen:

- Wat heeft LIGHTNING te bieden en hoe functioneert het?,
- Hoe worden programma's erdoor (positief) beïnvloed,
- Wat zijn de overeenkomsten/verschillen met zijn naaste concurrent SPEEDSCREEN,
- Problemen met LIGHTNING en
- een conclusie.

Eigenlijk is LIGHTNING een van de programma's waarover weinig valt uit te leggen. Zoals de advertentie van Digital Precision al zegt kan zelfs een aap LIGHTNING installeren met behulp van het configurator programma.

Het gebruik van LIGHTNING beperkt zich daarna tot het laden van de code bij het opstarten (LIGHTNING moet als eerste geladen worden, dus nog voor QRAM of SPEEDSCREEN, hoewel dit laatste weinig zin heeft daar LIGHTNING en SPEEDSCREEN ongeveer hetzelfde doen).

LIGHTNING blijft volledig onzichtbaar voor de gebruiker.

Het enige verschil is dat alle scherm uitvoer - tekst EN graphics - een stuk sneller gaat en bovendien de meeste mathematische functies (SIN, EXP, SQRT enz.) een stuk zijn opgevoerd. LIGHTNING voegt nog een aantal commando's toe aan QDOS die allemaal beginnen met '\_Ing' (persoonlijk vindt ik de underscore (\_) een beetje onhandig, vooral aan het begin van keywords. In de praktijk heb je deze commando's nauwelijks nodig behalve wanneer je het uiterste, wat de snelheid betreft, uit LIGHTNING wilt persen.

## Inleiding

We weten allemaal dat de QL nooit de 100 meter binnen de 10 seconden zal lopen. Toch worden er nog steeds verwoede pogingen gedaan om die zwarte doos wat op te peppen. Wie herinnert zich nog de SuperBASIC interpreter die op zijn doolie gemak de programmaregels onder handen nam? Na de komst van Supercharge, Turbo en Q\_Liberator zijn we van dit lijden goeddeels verlost.

Toch bleef er het een en ander over wat zelfs de compilers niet oplosten. Zo is het printen van tekst nog steeds een langdurige geschiedenis. De regels kruipen over het scherm en dit wordt niet versneld door compileren omdat de compilers dezelfde routines in ROM aanspreken als de interpreter. Deze routines om tekst op het scherm te printen zijn in de opeenvolgende ROM-versies niet gewijzigd. Bij het ontwikkelen van de allereerste ROMs hadden de programmeurs nogal te kampen met ruimtegebrek. Daarom zijn de print-routines zeer compact gehouden wat ten koste ging

van de snelheid. Dit betekende dat er een routine in ROM zit die het afdrukken van een karakter verzorgt. Hierbij wordt dan rekening gehouden met de display-mode, de karaktergrootte (CSIZE), de STRIP-kleur, de INK-kleur, de UNDER-status, de OVER-status en de positie op het scherm. Je begrijpt wel dat, voordat dit allemaal is uitgezocht, er behoorlijk wat tijd is verstreken.

Nu tegenwoordig bijna iedere QL-gebruiker geheugen genoeg heeft, konden de tekst routines eens wat ruimer opgezet worden. Degene die dit gat in de markt het eerst zag was Simon N. Goodwin van "Creative Codeworks" (voormalig Digital Precision). Zijn bekende SPEEDSCREEN gaf een enorme verbetering van het behandelen van tekst op het scherm. Dit moest je echt zien om te kunnen geloven. In het gunstigste geval gaf SPEEDSCREEN een snelheidswinst van 12x!

Bij Digital Precision hebben ze het er niet bij laten zitten en sinds enige tijd is LIGHTNING op de markt gekomen. Deze code versnelt niet alleen de tekst maar voegt nog een aantal extra routines toe aan QDOS. Deze routines vervangen de grafische procedures en een aantal mathematische functies. Niet alleen de tekst wordt dus versneld, ook het tekenen van lijnen en cirkels en het uitrekenen van bijvoorbeeld sinussen en vierkantswortels gaat een stuk sneller. Ik ga op elk van deze aspecten nog uitgebreid in.

LIGHTNING bestaat uit 3 componenten die ieder op zichzelf staan en ook afzonderlijk gebruikt kunnen worden. Heb je echter voldoende geheugen dan kun je ze alle drie tegelijk bij het opstarten in de RESIDENT PROCEDURE AREA binnenladen met de gebruikelijke BOOT-procedure:

```
base=RESPR (space): LBYTES  
Ing_TEXT_ext,base : CALL base
```

Dit laadt bijvoorbeeld alleen de TEXT-



enhancement routines binnen.

Er is overigens een programma bij LIGHTNING gevoegd dat vrijwel automatisch een BOOT-file aanmaakt waarin alle extensies, die nodig zijn, worden geladen. Zijn de extensies eenmaal met QDOS 'gelinkt' dan heb je er verder geen omkijken meer naar. De standaard QDOS tekst-, grafische- en mathematische- routines worden door LIGHTNING overgenomen en zo goed mogelijk gemiteerd met slechts minimale verschillen, alleen ... veel sneller!

---

### de text-enhancements

---

Op ongeveer dezelfde manier als in SPEEDSCREEN worden de routines voor tekst in CSIZE 0,0 en CSIZE 0,1 (dus alleen in MODE 4) onder handen genomen. Over het snelheidsverschil straks meer. met de nieuwe commando's \_IngON en \_IngOFF kan de werking van LIGHTNING aan, respectievelijk uit gezet worden (ook wanneer QRAM geladen is, bij SPEEDSCREEN gaf dit problemen). Normaal gesproken hoeft dit niet gebruikt te worden omdat de tekst routines automatisch aangezet worden bij het opstarten. Je merkt van de tekstversnellers overigens niets behalve dat alle tekst een stuk sneller geprint wordt. Alles gebeurt in de achtergrond en op precies dezelfde manier als de QDOS-code zou doen. LIGHTNING heeft effect op alle jobs tegelijk. Dus ook op QUILL, The Editor en andere jobs die in de QL draaien.

Behalve de 'schrijf een karakter naar het scherm' code worden ook het 'clearen' van het scherm en het PANNen en SCROLLen nog versnelt. Wanneer tekst een window helemaal gevuld heeft wordt normaal gesproken een regel opgeschoven. Daar dit scrollen veel tijd kost ten opzichte van het schrijven van tekst, kan een tijdswinst behaald worden door meerdere regels tegelijk te scrollen. Met het commando \_IngZIP kan dit aantal regels ingesteld worden. Een file kan op deze manier door een window gejaagd worden. Vooral wanneer je veel regels tegelijk opschuift: met \_IngZIP 10 kun je in een paar seconden een file de revue laten passeren. \_IngZIP heeft echter geen effect op programma's die het scrollen van windows zelf controleren zoals de SuperTool-

kit II editor ED en QUILL.

Een extra toevoeging aan het systeem is een nieuw device, het zogenaamde 'null'-device. Dit device, dat vergelijkbaar is met b.v. het ser-device, kan alleen voor uitvoer gebruikt worden. Alleen, die uitvoer komt nergens op uit en verdwijnt als het ware in een zwart gat. Dit kan wel eens van pas komen om ongewenste uitvoer op een snelle manier te laten verdwijnen. Want het null-device is het snelste device dat er voor de QL bestaat. Het is zelfs veel sneller dan een ramdisk.

Het nul-device kan op ieder moment gebruikt worden met de toetscombinatie CTRL/Z. Als je dit indrukt wordt alle tekst-uitvoer op het scherm naar dit zwarte gat gestuurd. Je ziet dan dus niets meer op het scherm verschijnen totdat een willekeurige toets wordt ingedrukt. Wil je bijvoorbeeld alleen het laatste gedeelte van een file zien, ga dan als volgt te werk:

```
copy flp1_file to scr_ <ENTER>
<CTRL/Z>
```

Druk na een paar seconden op SPATIE en de tekst komt weer op het scherm. Herhaal zonodig <CTRL/Z> als je nog niet bij het gedeelte bent dat je wilt bekijken. Pas wel op dat niet alles in het afvalputje verdwijnt (bedenk dat het null-device supersnel is)!

Indien nodig kan met \_IngKEYSET de control-toets veranderd worden. Persoonlijk vind ik de Z-toets wel een handige keuze.

Net zoals bij SPEEDSCREEN worden ook bij LIGHTNING een aantal (23!!) fonts bijgeleverd. Deze kunnen met het nieuwe commando \_IngFONT gestalleerd worden. Voor de karaktergrootten CSIZE 0,1 zijn er speciale fonts die van alle 8 kolommen in een 0,1 karakterblok gebruik maken. Er kunnen wat problemen optreden met deze brede fonts: wanneer er namelijk in dezelfde gereserveerde geheugenruimte een ander font wordt geladen (dus over een CSIZE 0,1 font heen) dat voor hetzelfde kanaal gebruikt wordt (met \_IngFONT #ch, fontadres) zonder dat tussentijds een karakter naar een ander kanaal in de CSIZE 0,1 grootte wordt geschreven, dan wordt het nieuwe font voor het betreffende kanaal genegeerd, d.w.z. het oude font blijft nog actief. Wie dit nog kan volgen weet vast wel raad

met deze situatie, in het onwaarschijnlijke geval dat ze mocht voorkomen!

Voor diegenen die geheugen te kort komen is er nog een verkleinde versie van de tekst routines. Deze versie versnelt de CSIZE 0,1 niet en het meervoudig scrollen werkt ook niet. Dit betekent dat de CSIZE 0,1 tekst met de QDOS-routines wordt afgehandeld. Dit is een van de voordelen van LIGHTNING: als een bepaalde functie niet wordt ondersteund of wanneer er geen snelheidswinst valt te behalen, dan geeft LIGHTNING de controle aan QDOS over. Er ontstaan dus nooit rare dingen op het scherm of erger: het crashen van de QL.

---

### de graphic enhancements

---

De QL bezit een uitgebreide reeks grafische routines waarmee het mogelijk is om mooie resultaten te verkrijgen. Helaas is de snelheid in vergelijking met andere computers betrekkelijk laag. LIGHTNING biedt ook hier weer uitkomst. Alle commando's die met het tekenen van punten (POINT, POINT\_R) het trekken van lijnen (LINE, LINE TO, LINE\_R en ook het turtle commando MOVE) en het tekenen van cirkels (CIRCLE en CIRCLE\_R) te maken hebben, zijn vervangen door snellere routines. Met uitzondering van de ARC routine wordt dus de hele grafische truckendoos opnieuw gecomplementeerd (merk op dat het BLOCK commando eigenlijk een buitenbeentje is, het werkt op de pixelcoördinaten i.p.v. de grafische; bovendien was dit commando al zeer snel).

De snelheidswinst van dit gedeelte van LIGHTNING is niet zo dramatisch als bij de tekstbehandeling maar is toch zeker de moeite waard. Ze varieert van 1.5 maal sneller voor cirkels tot 4 maal voor het POINT commando. De graphic enhancements werken zowel in MODE 4 als MODE 8. Zoveel mogelijk is getracht om de QDOS-code te simuleren maar af en toe zit LIGHTNING er wel eens een pixel naast. In de meeste gevallen zal dit geen problemen opleveren daar het weinig uitmaakt of een cirkel of lijn nu wel of niet door een bepaald pixel gaat. Het kan wel problemen geven wanneer de grafische routines van



LIGHTNING uit- of aan-gezet worden (met het \_IngGRAF commando) wanneer in OVER -1 mode wordt gewerkt. Wordt dan een lijn nogmaals over een reeds getrokken lijn gezet terwijl LIGHTNING net is uitgeschakeld dan zullen er een paar pixels of zelfs een hele cirkel op het scherm achterblijven.

Probeer dit maar eens:

```
PAPER 0:INK 7:OVER -1:CLS { maak
het scherm leeg }
_IngGRAF 1 { zet LIGHTNING aan,
voorzover dit nog niet gebeurd was. }
LINE 0,0 TO 50,100 { teken een dia-
gonale lijn ... }
LINE 0,0 TO 50,100 { ... en wis hem
weer uit }
LINE 0,0 TO 50,100 { teken hem nog-
maals ... }
_IngGRAF 0 { .... LIGHTNING uit ... }
LINE 0,0 TO 50,100 { .. en er blijven
nu 'n stel pixels staan !! }
```

Gebruik nu eens het commando CIRCLE 50,50,40 i.p.v. LINE. Je zult zien dat er de tweede keer twee cirkels blijven staan!

Remedie: laat LIGHTNING of aan, of uit staan.

Een pluspunt van LIGHTNING is dat een aantal bugs of op zijn minst eigenaardigheden van het grafische pakket worden opgelost. Type maar eens in (met LIGHTNING uit):

```
MODE 8: FILL 1
FOR i=0 TO 100 STEP 2:
INK i:CIRCLE 50,50,40
```

en je zult zien dat een nieuwe gevulde cirkel in de linkeronder hoek niet altijd precies de vorige overlapt. Een ander voorbeeld:

```
CIRCLE -500,50,550
```

moet theoretisch een gedeelte van een cirkel te zien geven. QDOS faalt hier maar LIGHTNING laat hier wel een boog zien zoals het hoort.

## De maths enhancements

Nog een gedeelte in QDOS dat om versnelling vraagt: de mathematische routines. LIGHTNING voorziet in weer een aparte file waarmee deze functies versneld kunnen worden. De vol-

gende functies zijn hierin opgenomen:

```
EXP LN LOG10 SIN ASIN
COS ACOS TAN ATAN COT
ACOT SQRT
```

Er wordt door Digital Precision geclaimd dat deze functies in snelheid verdubbeld worden zonder een verlies in precisie. Hierover zo meten wat meer. Bovendien wordt het domein van de functies EXP, LN en LOG10 uitgebreid. PRINT EXP(600) geeft normaal een 'overflow' maar lightning gaat nog door tot voorbij EXP(1400) (=1.028667E608). Niet dat je hier in het dagelijks leven zo veel aan hebt, maar als je eenmaal een exponent van 12 bits hebt, moet je deze ook kunnen gebruiken (hebben ze bij DP gedacht).

De verdubbeling in snelheid is al heel wat maar met LIGHTNING kan nog meer bereikt worden.

Normaal gesproken heeft een floating point getal een mantissa van 4 bytes (=32 bits, 1 lang woord).

Nu kan met een speciaal commando (\_IngPREC) dit aantal bytes verlaagd worden ten koste van de reken nauwkeurigheid.

De parameters voor \_IngPREC x zijn:

4 (default), reken met dezelfde precisie als QDOS, echter 2x zo snel. Met deze instelling zullen de meeste programma's geen hinder ondervinden van LIGHTNING, alleen maar plezier.

2 Gebruik een mantissa van 2 bytes. De snelheid is weer een factor hoger als bij x=4

1 Een 1 byte mantissa wordt gebruikt wat wil zeggen dat de uitkomst van de functies slechts 2.5 significante cijfers bevat. Deze instelling kan sommige programma's, die afhangen van de 9-cijfer precisie van QDOS, dainig in de war brengen. U bent gewaarschuwd! Maakt de precisie niet veel uit dan kan het rekenen met deze instelling een enorme snelheidswinst opleveren (tot 10x sneller).

0 Hierdoor worden de LIGHTNING mathematische routines uitgeschakeld en neemt QDOS alles weer voor zijn rekening.

Het gebruik van \_IngPREC 1 heeft de volgende consequenties (behalve het verlies in precisie):

- De functies LN, LOG10, EXP, COT en ACOT worden bij deze instelling toch met 2-byte precisie uitgerekend;
- De andere functies, behalve SQRT, worden alleen binnen een bepaald domein-interval uitgerekend:

functie	interval voor argument
SIN	- $\pi/2$ tot $+\pi/2$
COS	- $\pi$ tot 0
TAN	- $\pi/4$ tot $+\pi/4$
ASIN	-1 tot +1
ACOS	-1 tot +1
ATAN	-1 tot +1

Valt een argument toch buiten een van deze intervals dan wordt de functie met de 2-byte precisie uitgerekend. Dit is weliswaar iets langzamer, maar geeft een resultaat dat beter in de buurt komt als wanneer het met 1-byte precisie werd uitgerekend. Deze ingebouwde veiligheid is tenslotte beter dan een snellere functie met een verkeerde uitkomst.

Er zit een verschil in het gebruik van de versnelde functies ten opzichte van de andere twee componenten van LIGHTNING: de tekst- en de grafische-routines. Terwijl de laatste twee componenten echt onzichtbaar blijven voor de gebruiker, zijn de nieuwe functies dat niet.

Voor de ingewijden: in de name-table worden de namen SIN, COS, EXP enz. vervangen door dezelfde namen waarbij een andere pointer hoort die wijst naar de LIGHTNING code i.p.v. naar de code in ROM. De Basic interpreter denkt dus dat het b.v. de normale SINUS functie gebruikt, terwijl de LIGHTNING code zijn werk doet.

Dit betekent dat alleen programma's en jobs, die van de name-table gebruik maken, van LIGHTNINGS snelheidswinst kunnen profiteren.

Gelukkig doen programma's die gecompileerd zijn met een van de compilers (Supercharge, TURBO en Q\_Liberator) dit. Machinecode- en pascal-programma's doen dit hoogst waarschijnlijk(?) niet en daarvoor hebben deze functies geen zin.



Zoals alle commando's, die gedeelten van LIGHTNING aan en uitzetten (\_IngON, \_IngOFF, \_IngGRAF 0/1), werkt ook \_IngPREC globaal. Dat wil zeggen dat alle jobs erdoor beïnvloed worden. Een uitzondering is wanneer een job zelf probeert om van \_IngPREC 1, 2 of 4 naar \_IngPREC 0 te gaan. Dit werkt niet zonder meer.

LIGHTNING lost, nu er toch het een en ander opnieuw geprogrammeerd wordt, de COTangens bug op. Deze bug, die in de QL World van Juni 1988 wordt genoemd, betreft de cotangens van 0. In feite is deze niet gedefinieerd en moet dus een overflow error geven. In plaats daarvan geven alle ROMs het foute antwoord "1". LIGHTNING geeft in dit geval een overflow error.

Voor de programma ontwikkelaars heeft lightning ook nog wat in petto: De library met snelle mathematische routines. dit zijn de bovengenoemde functies die in basic programma's gebruikt kunnen worden met een F als voorvoegsel. Dus SIN wordt FSIN, EXP wordt FEXP enz. De snelheid en nauwkeurigheid is dezelfde als van de normale LIGHTNING extensies en met het commando \_IngFPREC kan op een soortgelijke wijze de nauwkeurigheid ingesteld worden.

Deze routines staan eigenlijk los van de rest van LIGHTNING en zijn bedoeld om door programmeurs gebruikt te gaan worden. Deze routines mogen zonder betaling aan DP in commerciele software gebruikt worden, zolang DP en LIGHTNING maar in de documentatie genoemd worden.

Een ander extraatje is het boven genoemde null-device. In SuperBASIC kan dit gebruikt worden op de volgende wijze:

OPEN #3, nul

Alles wat nu naar kanaal 3 wordt gestuurd verdwijnt daarin om nooit meer terug te komen. Dit gebeurt wel verschrikkelijk snel. In sommige gevallen kan zo'n afvoerputje wel eens van pas komen, bv. om ongewenste uitvoer snel onder de mat te werken. In andere gevallen kan het het programmeren wat eenvoudiger maken. Het voert te ver om hier gedetailleerd op in te gaan.

## Benchmarks

Misschien, en ook wel terecht, bent u alleen genteresseerd in het nu volgende: wat levert het allemaal op; hoe snel is LIGHTNING nu eigenlijk? Daarom heb ik een aantal Benchmarks uitgevoerd die het verschil aangeven tussen de standaard routines, LIGHTNING en de laatste versie van SPEEDSCREEN die in mijn bezit is (1.26). Er kan veel voor en tegen benchmarks gezegd worden en het is vrij simpel om de feiten te verdraaien aan de hand van zogenaamde objectieve tests. Ik heb hier geprobeerd om zo eerlijk en realis-

tisch mogelijk de enhancement pakketten tegenover elkaar te zetten.

Er kunnen verschillen zitten in de benchmark-resultaten ten gevolge van een andere systeem configuratie. Deze verschillen mogen wel absoluut zijn maar niet relatief! Deze tests zijn uitgevoerd op een JS-QL met 512k Miracle expanderam, diskinterface en SuperToolkit II.

Allereerst zijn er bij SPEEDSCREEN en LIGHTNING benchmarks gevoegd die natuurlijk geoptimaliseerd zijn voor de nieuwe tekst routines. Deze beide benchmarks heb ik voor beide versnellers gebruikt. Het resultaat staat in de volgende tabellen:

### SPEEDSCREENS "printspeed\_bas"

						tijden in (seconden)	
INK	OVER	UNDER	PAPER	CSIZE	QDOS	SPSCR 1.26	LIGHTNING
7	0	0	0	1	105	15	12
128	0	0	150	1	105	21	21
16	0	1	102	0	134	134	41
2	1	0	-	0	122	35	30
2	-1	0	-	0	123	31	26
2	0	0	4	0	133	42	41
2	0	1	4	0	134	44	41
0	0	0	7	0	133	42	41
0	0	0	7	1	105	21	21

Hieruit volgt dat SPEEDSCREEN het printen van tekst met gemiddeld een factor 3.87 versneld terwijl LIGHTNING gemiddeld een factor 4.50 haalt. Met andere kleurcombinaties zal hier misschien een klein verschil in deze factoren optreden maar ik heb voor zowel LIGHTNING als SPEEDSCREEN de 'best-' en 'worst-' case uitgetest. LIGHTNINGS "demo\_text\_bas" De test van LIGHTNING is wat eerlijker en betrouwbaarder omdat alle STRIP, INK, OVER en UNDER combinaties precies n keer getest worden en omdat de testperiode ruim een uur bedraagt. De resultaten zijn als volgt:

	seconden	versnelling [ % ]
QDOS	5045-1225=3820	0.0 %
SPEEDSCREEN v1.24	4769-1225=3544	7.8 %
SPEEDSCREEN v1.26	2578-1225=1353	182.3 %
LIGHTNING	2573-1225=1348	183.4 %

De 1225 seconden die van de tijden wordt afgetrokken komt van de interpreter overhead. Om het werkelijke verschil tussen de test-tijden te kunnen meten moet alleen naar de tijd dat geprint wordt gekeken worden. Uit deze test blijkt dat de versie 1.26 van SPEEDSCREEN nauwelijks onderdoet voor LIGHTNING. De 7.8% versnelling van het 'oude' SPEEDSCREEN lijkt weinig maar dit komt omdat een heleboel kleurcombinaties door deze SPEEDSCREEN niet versneld worden. Dit is op zich geen ramp omdat deze stipules in de praktijk nauwelijks voorkomen.



De volgende benchmarks hebben alleen betrekking op LIGHTNING. SPEEDSCREEN versneld namelijk alleen de tekst routines. LIGHTNING vervangt ook nog de meeste mathematische functies en grafische routines. De benchmark programma's op zich zijn tamelijk nutteloos. Toch is geprobeerd om iets te programmeren, met gebruik van de versnelde functies, dat in een serieuzer (teken?) programma kunnen voor zou kunnen komen.

## ROTATE

Dit programma maakt gebruik van de routines LINE en de functies SIN en COS om een rechthoek in het X-Y-vlak te laten roteren. Dit soort code zou, aangepast, ook in een CAD-programma voor kunnen komen. Het programma op zich kan nog wel geoptimaliseerd worden maar dat is met opzet niet gedaan.

Bedoeling was om zo veel mogelijk de genoemde functies uit te voeren.

```
100 REMark ROTATE BLOCK
110 :
120 PAPER 0:INK 7:SCALE
    100,50,-50:OVER -1 :
    BORDER 1,2:CLS
130 :
140 DIM c(3,1),nc(3,1)
150 REPEAT loop
160 PRINT
170 INPUT '_lnggraf?'
    (0/1)!'graf: _lngGRAF graf
180 INPUT '_lngprec? (0/4/2/1)!'prec:
    _lngPREC prec
190 CLS
200 RESTORE
210 FOR i=0 TO 3
220 FOR j=0 TO 1:READ c(i,j)
230 END FOR i
240 DATA -10,-40, 10,-40
250 DATA 10, 40, -10, 40
270 t1=DATE
280 FOR a=0 TO -PI/4
    STEP -1E-2
290 box a
300 box a
310 END FOR a
320 time=DATE-t1
330 PRINT 'TIME:!'time!'secs'
340 END REPEAT loop
350 :
360 DEFine PROCedure
    box(angle)
370 LOCAL i
```

```
380 FOR i=0 TO 3
390 nc(i,0)=c(i,0)*COS(angle)-c
    (i,1)*SIN(angle)
400 nc(i,1)=c(i,0)*SIN(angle)+c
    (i,1)*COS(angle)
410 END FOR i
420 draw_figure
430 END DEFine box
440 :
450 DEFine PROCedure draw_figure
460 LOCAL i
470 FOR i=0 TO 2
480 LINE nc(i,0),nc(i,1) TO nc
    (i+1,0),nc(i+1,1)
490 END FOR i
500 LINE nc(3,0),nc(3,1) TO nc
    (0,0),nc(0,1)
510 END DEFine
```

Afhankelijk van de \_lngGRAF en \_lngPREC instellingen wordt een verbetering van 3 tot 126 procent verkregen. Zet je in regel 290 en 300 'FILL 1' voor 'box a' dan wordt het resultaat nog beter. Ook de FILL functie is aanzienlijk versneld.

## FUNCTIONS

Hiermee worden voornamelijk alle versnelde mathematische functies n de POINT procedure getest op snelheid. Een wiskunde programma voor het analyseren van functies, afgeleiden enz. zou hier zijn voordeel mee

kunnen doen.

```
100 REMark Lightning test
110 :
120 WINDOW 448,200,32,16
130 REPEAT loop
140 INPUT #0,'lnggraf?','gr:
    _lngGRAF gr
150 INPUT #0,'lngprec?','pr:
    _lngPREC pr
160 CLS
170 t1=DATE
180 FOR i=-PI/4 TO PI/4 STEP
    PI/800
190 x=(i+PI/4)*100
200 POINT x,50+SIN(i)*40
210 POINT x,50+ASIN(i)*20
220 POINT x,50+COS(i)*40
230 POINT x,50+ACOS(i)*20
240 POINT x,50+TAN(i)*40
250 POINT x,50+ATAN(i)*20
260 POINT x,50+SQRT(ABS
    (i))*40
270 POINT x,60+LN(ABS(i))*10
280 POINT x,50+LOG10(ABS
    (i))*40
290 POINT x,EXP(i)*20
300 POINT x,60+COT(i)*1
310 POINT x,50+ACOT(i)*30
320 END FOR i
330 t2=DATE
340 PRINT #0,t2-t1,'secs'
350 END REPEAT loop
```

De testresultaten zijn in deze tabel opgenomen (de tijden zijn al aangepast; de overhead is 17 seconden):

_lngGRAF	_lngPREC	seconden	versnel. [%]	factor
0	0	118	0 %	1.00
0	4	107	10 %	1.08
0	2	102	15 %	1.13
0	1	100	18 %	1.15
1	0	50	136 %	2.01
1	4	40	195 %	2.37
1	2	35	237 %	2.60
1	1	33	257 %	2.70

Hieruit blijkt dat vooral wanneer de LIGHTNING graphics worden ingeschakeld, de snelheid sterk toe neemt. De POINT routine is zeer snel geworden.



## SPIRAL

Spiral is iets minder serieus. Het laat zien dat met de turtle commando's in een paar programmeerbare regels mooie (en met LIGHTNING ook snelle) resultaten verkregen kunnen worden. Probeer dit maar eens uit met OVER - 1 en FILL 1 !!

```
100 REMark TURTLE example
110 :
120 WINDOW 512,256,0,0
130 SCALE 100,0,0:FILL 0:
    PAPER 0:INK 7
140 POINT 70,50
150 PENDOWN: TURNT0 0
160 INPUT #0,'Over?
(0,-1)',ovr:OVER ovr
170 INPUT #0,'Fill? (0, 1)',
    fil: FILL fil
180 CLS
190 t1=DATE
200 FOR i=1 TO 180 STEP .25
210 MOVE i: TURN 91
220 END FOR i
230 PRINT #0,DATE-t1,'secs'
```

## Problemen

Tijdens het gebruik van LIGHTNING ben ik de volgende problemen (bugs) tegengekomen:

- In de precisie 1- en 2-byte precisie ( \_IngPREC 1/2 ) kan de QL crashen bij het uitrekenen van de tangens van 3.14159263 (=1-2.42e-8),

- De functies LN, LOG10 en COT worden zeer onbetrouwbaar wanneer met 1-byte en 2-byte precisie wordt gewerkt (de COT alleen voor argumenten < 0). De LN en LOG10 geven voor enkele waarden totaal verkeerde uitkomsten terwijl de rest van de functie vrij nauwkeurig is. Zie hiervoor ook het programma FUNCTIONS.

- De FILL 1 functie is wel sneller, maar raakt in de war bij ingewikkelde figuren (wanneer de interne FILL buffer te groot wordt). Waar de bug precies zit weet ik niet. Meestal crasht de QL pas na een tijdje omdat er systeem informatie wordt overschreven (common heap headers volgens

Voor deze test moet vooraf LIGHTNING aan- of uit-gezet worden met het \_IngGRAF commando. De tijden voor dit programma:

_IngGRAF	OVER	FILL	MODE 4	MODE 8
0	0	0	55	47
0	0	1	132	112
0	-1	0	54	46
0	-1	1	140	119
1	0	0	34	29
1	0	1	75	65
1	-1	0	33	28
1	-1	1	82	72

Het blijkt dat de typische versnelling die LIGHTNING voor dit programma te bieden heeft tussen de 1.6 en 1.8 maal ligt.

mij). Voor de 'gewone' vul-operaties van eenvoudige figuren, cirkels, vierkanten e.d. geeft FILL geen problemen.

## Conclusie

Na het uitvoerig testen van LIGHTNING en het gebruik ervan gedurende een paar maanden ben ik zeer te spreken over deze 'onzichtbare' software.

Wat de tekst uitvoer betreft is LIGHTNING iets beter als SPEEDSCREEN, de graphics worden niet zoveel opgevoerd maar zijn toch zeer de moeite waard. De getallenkrakers zullen veel plezier hebben van de opnieuw gementeerde functies.

Als je nog geen SPEEDSCREEN of LIGHTNING hebt dan verdient LIGHTNING de voorkeur.

Heb je SPEEDSCREEN reeds gekocht en werk je voornamelijk met tekst (de psion pakketten) dan heeft het niet veel zin om ook nog eens LIGHTNING aan te schaffen.

Overigens worden de functies in ABACUS en EASEL en de meeste graphics in EASEL met LIGHTNING niet versneld.

Marcel Kamp

## Te koop aangeboden:

QL JM 128K, Philips monitor, org. QL-Printer, 23 cartridges m/z software, QL boeken, Quasars compleet

Vaste prijs ..... f 1.000,-

G.F. Ploeg  
023 - 354119

## Te koop aangeboden

QL JS 384 KB  
Compleet met Sandy Super Q- Board, Monochrome Monitor, real time clock, TEAC dick-drive met voeding dit alles ingebouwd in een IBM - kast, met veel software op diskettes compleet met manuals

Compleet ..... f 900,-

Will Langezaal

Tel. 03402 - 64026



# EPROM PROGRAMMER VOOR ZELFBOUW

In dit artikel wordt een EPROM programmer besproken welke men zelf voor ca. 125 gulden kan bouwen. Ondanks deze vrij lage kostprijs heeft deze programmer welhaast professionele specificaties. Het is mogelijk EPROM's van 8K (2764) tot en met 64K (27512) te programmeren, volgens de standaard (50 ms puls) methode of de snellere "intelligente" methode, welke met 1 ms pulsen werkt. Bij deze laatste methode is de programmeersnelheid zo'n 10K per minuut.

Hoewel er geen printlayout is gegeven is het de bedoeling dat de programmer als een via de uitbreidingspoort aan te sluiten uitbreiding wordt gebouwd. De besturingssoftware bevindt zich zelf op EPROM in verband met mogelijke timing-problemen welke in RAM zouden kunnen optreden. De vereiste 12,5V of 21V programmeerspanning wordt door een omvormer in de programmer zelf opgewekt zodat geen externe voeding nodig is.

## DE HARDWARE

Het hart van de programmer wordt gevormd door een Intel 8255 parallel I/O poort. Deze heeft 24 lijnen beschikbaar, net voldoende dus om de 8 data- en 16 adreslijnen van de te programmeren EPROM aan te sturen. Omdat we echter nog wat extra besturingssignalen nodig hebben om de CE- en OE-aansluitingen te sturen alsmede de programmeerspanning, zijn nog vier extra lijnen gemaakt met een 74LS174 flipflop. Aangezien dit IC zes flipflops herbergt hebben we zelfs nog twee lijnen over, maar die worden nu nog niet gebruikt. De adresdecodering wordt verzorgd door IC3 en IC4. IC3 is een 8-bit comparator welke het juiste 16K-gebied selecteert uit de QL memory map. Aangesloten op een standaard QL zal dit

gebied altijd op adres \$C0000 beginnen, maar bij bijvoorbeeld een Thor zal dit anders zijn, afhankelijk van het adres op de lijnen SP0-SP3. Overigens voldoet het hele ontwerp aan de QL standaard voor wat betreft uitbreidingen, dus op een QL-kloon of een uitbreidingskaart behoort het ook te werken. IC4 selecteert binnen het 16K gebied de EPROM met de besturingssoftware (IC2, onderste 8K), de 8255 I/O (op \$2000) of de flipflops uit IC5 (op \$3000). De voeding bestaat uit drie delen: de 5V-voeding met de 7805 mag als bekend worden verondersteld, de 78L06 dient voor het verkrijgen van een 6V voedingsspanning nodig voor het "snel" programmeren, en tenslotte de omvormer met IC8 welke 12,5 of 21V aflevert. Dit bewuste IC (een TL497) bevat een complete schakelende regulator, met stabilisatie en stroombegrenzing. Voor de benodigde spanningsopslingering wordt een spoel van 1,5 mH gebruikt. De 12,5 of 21V programmeerspanning wordt via schakeltransistoren T2 t/m T5 naar de EPROM gevoerd. Deze schakeltransistoren maken softwarematig in- en uitschakelen van de programmeerspanning mogelijk. Wat wel handmatig geschakeld dient te worden is de voedingsspanning van de te programmeren EPROM (normaal 5V maar tijdens "snel" programmeren 6V) en de hoogte van de programmeerspanning (12,5V of 21V).

## HET BOUWEN

Zoals vermeld kan ik geen printlayout geven omdat ik zelf niet de benodigde spullen daarvoor in huis heb en het hier om een prototype gaat. Wellicht is hier een taak weggelegd voor een mede-Quasar lezer. Indien er in het schema geen pin-aansluiting bij een IC is aangegeven kunt u uit meerdere mogelijkheden kiezen (zie aansluitingen IC's). Bijvoorbeeld adreslijn A19 hoeft niet noodzakelijk aan aansluiting P7 van IC3 te liggen maar kan ook aan een andere P-ingang zitten, mits de corresponderende Q-ingang natuurlijk overeenkomstig Q7 in het schema is aangesloten!

Zo hebt u wat speling in de printlayout. Wat niet in het schema is aangegeven maar wel in de schakeling hoort te zitten zijn ontkoppelcondensatoren. Een 100 nF condensator per 1 a 2 IC's, verspreid over de print, is voldoende. Voor de diodes D5 en D6 zijn 1N4148 siliciumdiodes gebruikt. Hoewel deze in het prototype goed functioneren is het mogelijk dat de spanningsval over de diodes net iets te groot is waardoor het "1"-spanningsniveau op pin 1 en 22 van de EPROM net buiten de specificatie valt. In dat geval helpt wellicht het vervangen van de diodes door AA117 types. Voor de EPROM-voet kan men het beste een ZIF (Zero Insertion Force) voet nemen. Zijn niet goedkoop maar werken veel makkelijker dan een gewone IC-voet. Het afregelen van de programmeerspanning gaat als volgt:

1. Bepaal het adres waarop de EPROM-programmer zit (\$C0000 op een standaard QL); noem dit a.

2. Geef het commando POKE a+12288,12. 3. Meet de spanning op pinnen 1 en 22 van de EPROM-voet en regel met R5 de 12,5V spanning af (S2 op 12,5V) en met R7 de 21V



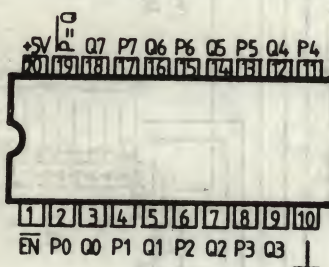
spanning (S2 op 21V). Regel zo nauwkeurig mogelijk af; minstens op 0,5V nauwkeurig maar 0,2V is beter. Het is ook mogelijk de spanning op pen 6 van IC8 te meten (is altijd gelijk aan de programmeerspanning) maar u dient dan ca. 0,2V hoger af te regelen vanwege de spanningsval over T4 en T5.

Het besturingsprogramma dat zich in EPROM IC2 bevindt wordt geactiveerd door vanuit BASIC het commando EPROM te geven. Van daaruit kunt u de diverse menu-gestuurde

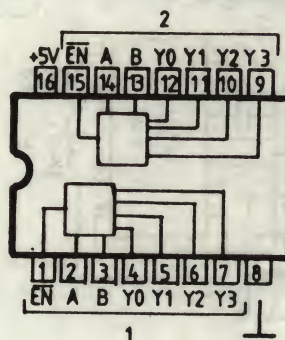
opties bereiken. Deze omvatten het inlezen van een programma van microdrive, disk of EPROM, editen van het ingelezen programma met een Hex/Ascii editor, wegschrijven van het programma naar MDV of disk, en natuurlijk het programmeren van de EPROM. Het is mogelijk reeds geprogrammeerde EPROM's te herprogrammeren, mits er alleen bits van "1" op "0" gezet dienen te worden (het omgekeerde is alleen mogelijk door de EPROM met ultraviolet licht te wissen). Er wordt automatisch vooraf en

achteraf gecontroleerd. U kunt in het bezit komen van de software en handleiding door een (bij voorkeur ongeprogrammeerde) 2764 EPROM te zamen met een cartridge of 3,5 inch disk (80 tracks) op te sturen naar onderstaand adres, MET retourporto! U ontvangt de geprogrammeerde EPROM dan retour tezamen met de handleiding op QUILL-doc.

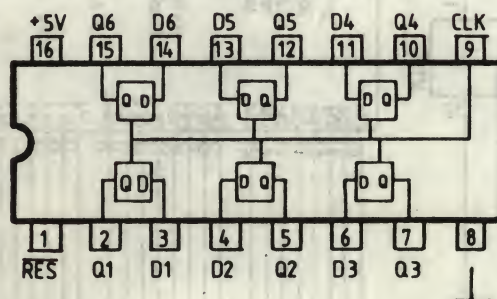
Jan Bredenbeek  
Diependaalselaan 255  
1214 KD Hilversum.



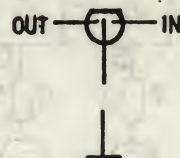
AM25LS2521 (74LS688)



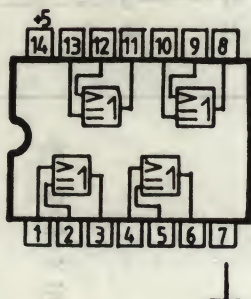
74LS139



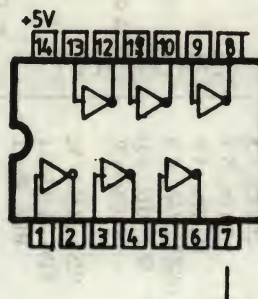
74LS174



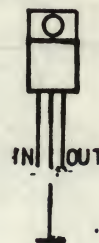
78L06  
(onderaanzicht)



74LS32



74LS04



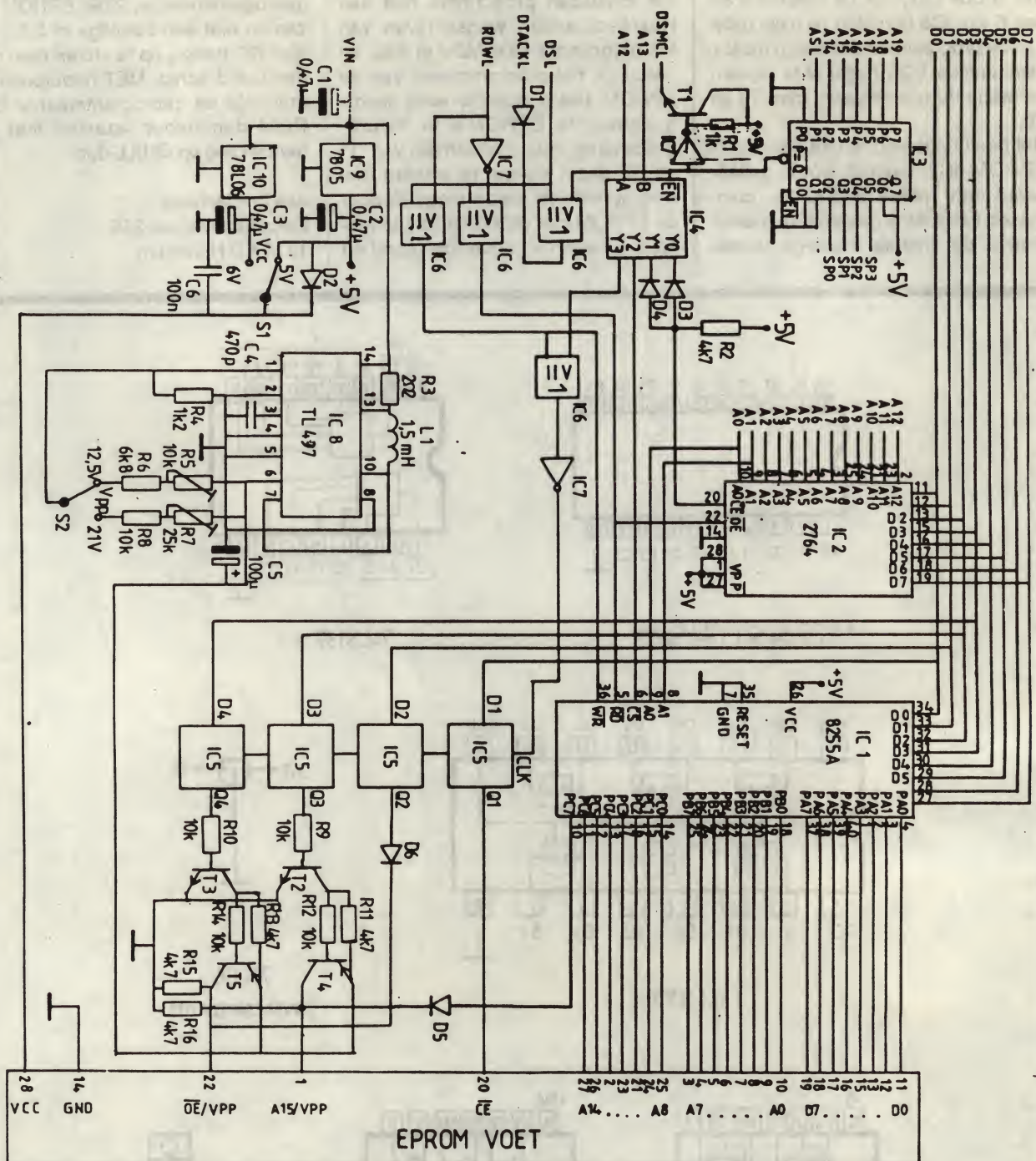
7805  
(vooraanzicht)



# EPROMmer

ISSUE 1 (4 - '88)

(C) 1988 J. Bredenbeek



EPROM VOET

- T1: BC 103B
- T2-T3: BC 547B
- T4-T5: BC 557B
- IC1: 8255A
- IC2: 2764 - 25
- IC3: AM25LS2521 (74LS688)
- IC4: 74LS139
- IC5: 74LS174
- IC6: 74LS32
- IC7: 74LS04
- IC8: TL497
- IC9: 7805
- IC10: 78L06
- D1...D6: 1N4148



# DATABANK-NIEUWS

**Het is alweer ruim een half jaar geleden dat ik iets over databanken in de Quasar heb geschreven, dus een "update" is inmiddels wel op z'n plaats.**

Allereerst iets over ons eigen Quasar BBS. De afgelopen tijd hebben veel mensen nogal wat problemen gehad bij het inloggen. Oorzaak daarvan was de nogal onorthodoxe methode waarmee het Astracom 1000 modem van de databank probeerde vast te stellen of er op 1200/75 baud of op 300/300 baud werd gebeld. Het is in de modemwereld een ongeschreven regel dat het opgebeldde modem reageert met het verzenden van een carrier (fluittoon) voor een bepaalde baudrate, als er na enkele seconden geen antwoord (carrier) van de opbeller komt wordt er enkele seconden lang een andere baudrate met een andere carrier geprobeerd enzovoort. Het Astracom modem echter verstuurt even een 1200/75 carrier, kijkt dan op de lijn wat er terugkomt, verstuurt dan weer een carrier enzovoort, met zeer korte tussenpozen.

En dat levert voor veel automatische modems nu problemen op, want die gaan pas online als ze ongeveer een seconde lang een stabiele carrier horen. Overigens zijn het juist de "slimme" en duurdere modems (Tandata, Hayes-compatible modems e.d.) die hier last van hebben, "domme" modems (Miracle en handgeschakelde modems) hebben geen problemen. Het probleem hebben we voorlopig even "opgelost" door het Astracom modem gewoon weer op alleen-1200/75 mode te zetten. Dan zijn er geen problemen meer bij het inloggen, maar is het uiteraard niet meer mogelijk op 300/300 in te loggen. Zo'n ramp zal dat vermoedelijk niet zijn want het gebeurt niet vaak dat iemand een file of bericht moet uploaden. Bezitters van modems die geen 1200/75 baud aankunnen staan nu echter wel in de kou. Er is op dit moment een mogelijkheid dat het Astracom modem wordt vervangen door een ander modem, dat behalve 300/300 en 1200/75 baud ook 1200/1200 baud full duplex (V.22) aankan. Dit vereist echter een hardware-ingreep in de databank-computer en het hangt van de welwillendheid van de sysop (Michel Spanjer) af of die

wordt uitgevoerd. Verzoeken daartoe kunt u kwijt aan de Quasar of op het BBS.

Het telefoonnummer is 03404-22533 en we zijn 24 uur per dag online. Met mijn eigen BBS (QBOX) gaat het goed. Zo goed dat ik in februari besloot een tweede telefoonaansluiting aan te vragen, waarop QBOX 24 uur per dag online zou kunnen zijn. Helaas zit de telefooncentrale in Hilversum echter vol en is er een wachttijd van vele maanden. En dus zit ik nu nog steeds te wachten op een nummer... Om toch enigszins aan de behoefte tegemoet te komen heb ik echter besloten de onlinetijd wat uit te breiden.

U kunt nu bellen: IEDERE DAG VAN 22.00 TOT 01.00 UUR.

Telefoonnummer: 035-42651; 300/300, 1200/75 en 1200/1200 baud; alleen BBS-mode (geen Viewdata!). Nogmaals dit geldt dus alleen van 22.00 tot 01.00 uur! Buiten deze tijden is dit telefoonnummer gewoon een spraaklijn en als u toch met een modem belt bezorgt u mij en mijn huisgenoten alleen maar een hoop ergernis en bent u 15 cent voor niets kwijt...

Jan Bredenbeek.

## Een klein kopje toe

De mogelijkheid bestaat dat schizofrenen tenminste twee keer zoveel vergissingen begaan als gewone mensen (computers uitgezonderd); in elk geval stonden er in het stukje "Twee koppen.." (mei 88, p.715) van mijn hand twee ondoordachte opmerkingen. Ze tasten de strekking van het verhaal overigens niet aan.

Jan Bredenbeek, die mij op een en ander wees, vertelde mij het volgende:

1) het feit dat er twee bytes gereserveerd zijn voor de lengte van een filename hangt samen met het QDOS-stringformaat (maximale stringlengte);

2) filetype 2 voor relocatables lijkt niet afhankelijk te zijn van de ROM-versie op de manier die ik aangaf: de macro-assembler van GST produceert bij een JM die aanduiding ook al (ik vraag me alleen af waarom The QDOS Companion het niet vermeldt en de Technical Manual wel).

Slecht doordachte en dito gecheckte opmerkingen schrap ik gewoonlijk - Quasar toch maar wat serieuzer nemen? Weer zo'n opmerking.

Bedankt Jan.

Ward Smid.

### Te koop aangeboden:

QL - JS.,512 K intern met reserve membraam, toolkit II  
Philips monitor 80  
CST diskinterfact en dubbele drives  
3.5" ( 720 K )  
vele cartridges en 20 diskettes.  
manuals e.d.  
Prijs f. 1200,-

Handelsondern. T.A.T.  
Nieuwe Singel 8  
3421 TA oudewater  
03486-2624



# SUCCESS Met CP/M

Deel 1

Wim van Dieren

Dit artikel is een bespreking van het programma **SUCCESS** van **DIGITAL PRECISION** alsmede een introductie van het operating systeem **CP/M**.

In een volgend artikel zal een toepassing van **SUCCESS** worden besproken nl: **dBASEII** en een vergelijk worden gemaakt met **ARCHIVE**.

Ik heb nu sinds 2 maanden regelmatig met het programma **SUCCESS** gewerkt en om meteen maar met de deur in huis te vallen het is een goede implementatie van het operating systeem **CP/M 80**.

Gedurende de testperiode heb ik geen programma's kunnen vinden welke het niet deden. Dit wil natuurlijk niet zeggen dat alle **CP/M** software feilloos draait maar mijn conclusie is dat het in ieder geval stukken beter werkt dan het programma van de concurrent **SANDY** met zijn programma **CP/Mulator**.

## WAT IS 'SUCCESS' ?

**SUCCESS** is een **CP/M** emulator (na-bootser) en dat doet het door een **Z80** processor na te bootsen. Het programma haalt de **Z80** instructies uit een gereserveerd stuk geheugen van **64K** van de **QL** en beschouwd deze instructies als **Z80** instructies. Het programma zal voor iedere **Z80** instructie een overeenkomstige instructie uitvoeren in de **68008** processor. In de tijd dat **CP/M** ontwikkeld werd was het meest gebruikte invoer/uitvoer apparaat een teletype (soort van veredelde typemachine) en waren beeldschermen en toetsenborden met functie en cursortoetsen nog toekomstmuziek. Het gevolg hiervan is dat **CP/M** niet voorziet in de aansturing van een beeldscherm. **SUCCESS** bootst daarom een **VT52** terminal na en deze kent via een serie escape codes cursor handelingen, directe cursor adresering, wist een scherm,

zet het display in inverse video of weer terug, wist cursor regel ect.

## WAT IS CP/M ?

**CP/M** is een operating systeem (besturings systeem). Een besturings systeem in het algemeen heeft de volgende functies:

1. In- en uitvoer
2. CPU beheer
3. Geheugenbeheer
4. Diskbeheer
5. Speciale commando's (utility's)

In feite is **CP/M** te vergelijken met **QDOS**, het overbekende operating systeem van de **QL**.

**CP/M** werd in 1974 ontwikkeld door **Digital Research** als een operating systeem voor 8 bit microprocessors en is vooral geïmplementeerd op machines met een **Z80** processor.

De versie welke door **SUCCESS** wordt nagebootst is **CP/M 2.2** In de loop der jaren is **CP/M** op zeer veel verschillende computers geïmplementeerd met helaas evenveel verschillende diskformaten.

Wat kan men met **CP/M** zoal doen ?

- Directory vragen met het **DIR** commando
- Het formateren met het **FORMAT** commando
- Het kiezen van een andere default drive
- De status van de disk opvragen met het **STAT** commando
- Het manipuleren van bestanden met het **PIP** commando

- Het weergeven op het scherm van een file met **TYPE**
- Het wissen van een bestand met **ERA**
- Het renamen van een bestand met **REN**
- Het editen van een bestand met **ED**

**CP/M** kent een aantal besturingstekens welke direct door het operating systeem worden herkend.

Deze tekens bestaan uit **CONTROL** codes aangeduid hier met **^** en een letterteken.

De meest gebruikte control codes volgen hieronder:

- ^C** warme start van **CP/M**
- ^P** uitvoer naar de printer aan/uit zetten
- ^R** drukt laatste regel opnieuw af.
- ^S** zet uitvoer aan/uit (vergelikbaar met **^F5** op de **QL**)
- ^U** verwijdert hele ingetypte regel en ga naar volgende regel
- ^X** verwijdert de ingetypte regel
- ^H** of de cursor links toets wist laatste teken
- ^I** of de tabulatie toets ga naar de volgende tab positie

## BIJGELEVERDE CP/M PROGRAMMA'S

De navolgende programma's worden met **SUCCESS** meegeleverd:

### ASM <filenaam>

Dit is een **8080** assembler welke een file maakt die vervolgens met het **LOAD.COM** programma omgezet kan worden tot een werkend **CP/M** programma.

### DDTZ <filenaam>

Dit is een machinetaal monitor programma wat ook kan disassembleren, assembleren, single stappen ect.

### ED <filenaam>

Dit is een primitieve tekstverwerker



en lijkt wel wat op het MS-DOS programma EDLIN.COM

#### LOAD <filenaam>

zet programma's in HEX formaat om in binaire vorm zodat deze gerund kunnen worden.

#### PIP<bestemming>= <sourcelijst>

Dit is een krachtig kopieerprogramma wat files kan kopiëren tussen diverse aangesloten apparaten en of disk.

Bijvoorbeeld: [PIP B:=A:\*.TXT]  
kopieert alle files met de extensie .TXT vanaf drive A: naar drive B:

#### STAT.COM

Een veelzijdig programma. het kan oa. de grote van files aangeven en kan files write-protecten, of uitvoer omleggen van bijvoorbeeld het scherm naar de seriele poort.

Voorbeeld: [STAT VAL:] laat alle opties zien voor het STAT commando (de [ ] staan er voor de duidelijkheid en moeten niet ingetypt worden!

#### SUBMIT.COM<tekst filenaam>

Dit programma kan een tekstfile met commando's uitvoeren alsof ze van het toetsenbord werden ingetypt. Het is vergelijkbaar met de MS-DOS batch file.

#### ANALYSE.COM

Dit is een utility geschreven in 68000 code en wordt gebruikt voor het analyseren van vreemde diskformaten zodat ze later rechtstreeks in CP/M gebruikt kunnen worden.

#### CAT.COM <device naam>

Geeft een directory van een QDOS schijf.

#### DEFINE <definitie file>

Wordt gebruikt om aan te geven welk vreemde disk formaat gebruikt moet worden (gebruikt de file aangemaakt door ANALYSE.COM).

#### DEFKEY.COM <toetscode>,'string' of <toetsnummer> of <toetscodefile>

Definieert de functie toetsen (zoiets als de ALTKEY toolkit2 commando maar dan voor CP/M).

#### DELQDOS.COM <qdosnaam>

Wist QDOS formaat files.

#### FORMAT.COM

Formateert een medium zodat het geschikt is voor CP/M.

#### RDQDOS.COM <cp/m naam> <qdosnaam>

Kopieert een QDOS file naar CP/M.

#### WRQDOS.COM <cp/m naam> <qdosnaam>

Kopieert een CP/M formaat file naar QDOS formaat.

#### SETBAUD.COM <baudrate>

Stelt de baudrate in van de seriele interface.

Bovenstaande is slechts een summier opsomming van wat de bijgeleverde programma's doen. Het zou te ver voeren om ieder programma in detail te bespreken. In het volgende deel zal ik in gaan op de installatie van de software en een beschrijving geven van het werken met CP/M in het bijzonder met dBASEII.

## ZELFBOUW ALTERNATIEF VOOR DE QTALKER

Nadat ik de prijs van de QTALKER had vernomen, zocht ik naar een spraaksynthesizer, waarvan de prijs redelijk is en de prestaties op het niveau van de QTALKER ligt.

In de E.T.I. van jan.1988 werd een artikel gepubliceerd van een RS232-Smarttalker. Dit bleek voor mij de oplossing. De Smarttalker is goedkoper en de prestaties liggen niet ver van elkaar. Het verschil is dat de QTALKER de tekens noemt als je een toets indrukt. Dit kan de smarttalker niet. Dit vond ik voor mijzelf niet onoverkomelijk. Voor de Smarttalker zijn diverse toepassingen te bedenken, o.a. het uitspreken van hele inge-

tikte listings; het uitspreken van een zin. Het voordeel is dat je bij het nakijken van een zelf ingetikte listing niet meer naar het scherm hoeft te kijken.

Voor de mensen die de Smarttalker willen nabouwen is een schema; onderdelenlijst; aansluitinggegevens en programma afgedrukt. De kosten van dit projekt bedraagt zonder behuizing met trafo etc. ongeveer Hfl.160,-- Voor de aansluiting tussen de Smarttalker en de QL is een RS232 kabel nodig, welke men kan bestellen bij Kees van der Wal.

Voor nadere informatie kunt U mij bellen. Veel succes met de bouw !!!!

Leo Moll 010-4263452  
(enkel na 18.00 uur)

#### Onderdelenlijst

R1,2 120R  
R3,4 10 K  
R5-9 68 K  
R10 1 K  
R11 10 R  
RV1 10 K instelpotmeter liggend model  
C1,C18,C19 100 uF 16V radiaal  
C2 1000 uF 16V radiaal  
C9 56 pF keramisch  
C3,6,7,8,13,14,16,17 100 nF keramisch  
C4,5 22 pF keramisch  
C10,11 10 nF keramisch  
C12,15 10 uF 16V radiaal  
CV1 55-65 pF trimmer  
IC1 7805  
IC2 74LS00  
IC3 CTS 256-AL2 (enkel bij TANDY te koop)  
IC4 74LS373



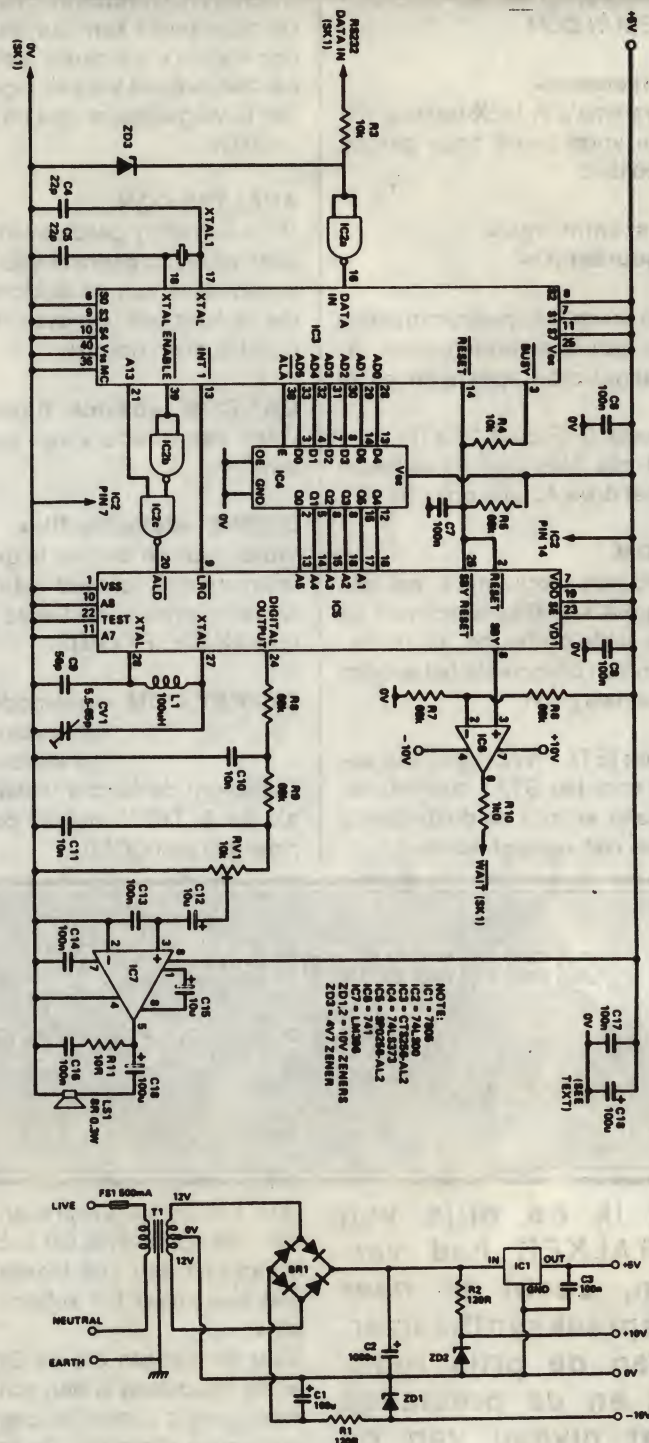
IC5 SP0 256-AL2  
 IC6 uA 741  
 IC7 LM 386  
 ZD1,2 10V 400 mW zenerdiode  
 ZD3 4V7 400 mW zenerdiode  
 BR1 type W004 of 1 Amp.brugcel  
 rond model  
 CON1 PCB connector bestaat uit 4  
 blokjes met elk 2 aansluitingen.  
 FS1 zekering snel 500 mA.  
 L1 100 uH. spoel  
 LS1 Luidspreker 8 Ohm 0.3 Watt.  
 SK1 3 pens DIN stekker  
 T1 transformator 12V-0V-12V  
 250 mA.  
 XTAL1 10 Mhz. kristal  
 Printplaat, RS232 kabel en koelplaat  
 voor IC1 10 x 10 cm.

#### AANSLUITGEGEVENS

QL SER1 Smarttalker  
 1 ground zwarte draad 0 Volt  
 2 TxD witte draad DATA IN  
 3 RxD groene draad niet aangesl.  
 4 DTR blauwe draad niet aangesl.  
 5 CTS rode draad WAIT  
 6 +12V oranje draad niet aangesl.  
 klipje connector

#### VOORBEELD VAN EEN PROGRAMMA

1 REMark this is an example listing  
 for the Smarttalker  
 2 REMark here follows the real listing  
 Listen  
 3 BAUD 300  
 4 OPEN#5,ser2  
 5 LIST#5  
 6 CLOSE#5  
 50 BAUD 300  
 60 OPEN#5,ser2  
 70 CLS:PRINT 'Om het programma te  
 beëindigen tik je als zinnetje  
 <STOP> in'  
 80 REPEAT talk  
 90 INPUT 'Tik een zinnetje in Engels  
 of het Nederlands',zin\$ 100 IF  
 zin\$='stop' OR zin\$='STOP' THEN  
 110 PRINT 'Dit was het dan !!' 120  
 EXIT talk  
 130 ELSE  
 140 PRINT#5,zin\$  
 150 END IF  
 160 END REPEAT talk  
 170 CLOSE#5



#### QL draadkleur

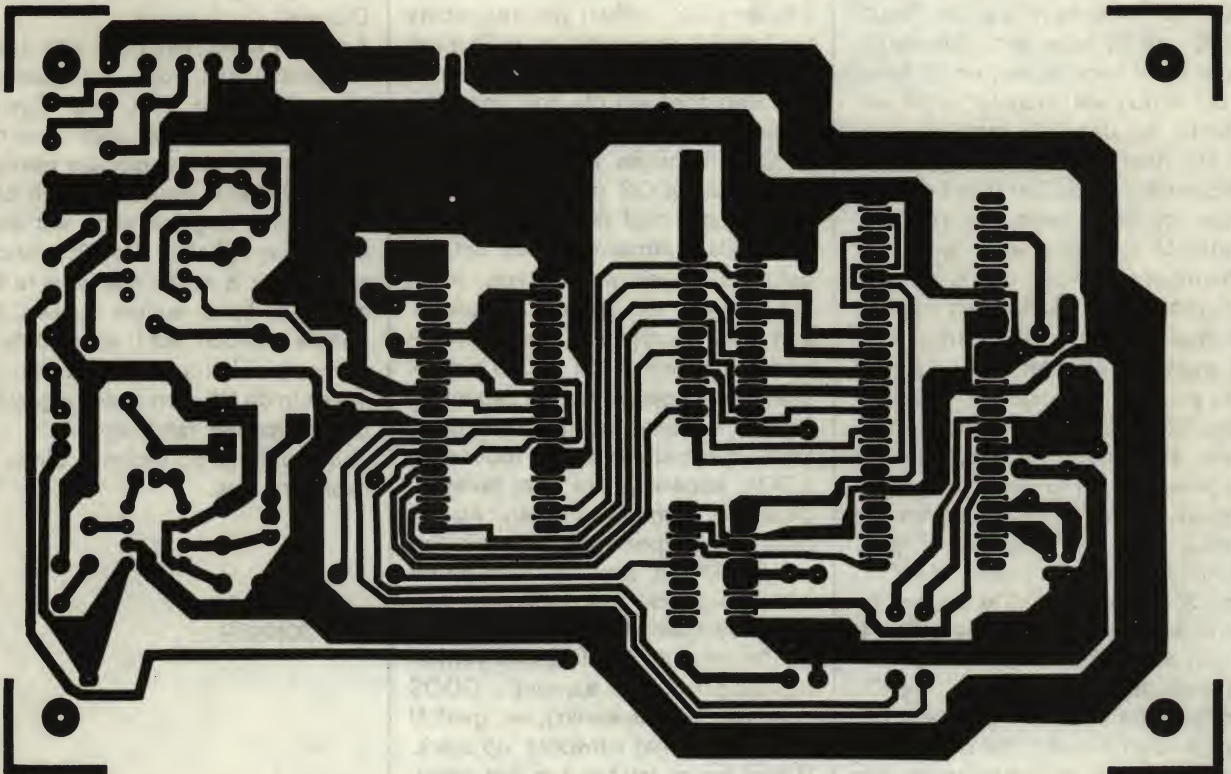
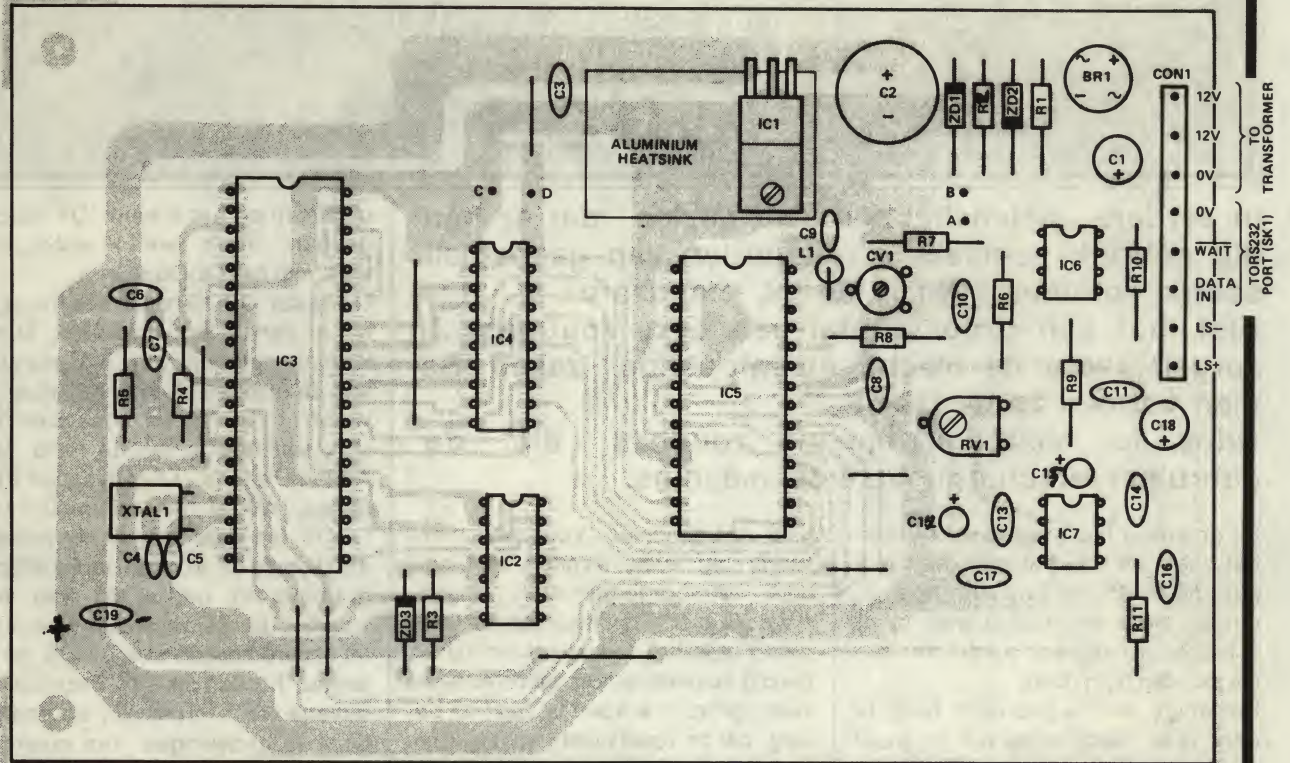
1	GND	zwart
2	TxD	wit
3	RxD	groen
4	DTR	blauw
5	CTS	rood
6	+12V	oranje

(clipje)

#### Smart Talker

12V	Tranfo
12V	
0V	
0V	al
0V	
WAIT	
DATA IN	LS
LS	







# CURSUS MACHINETAAL

## Hoofdstuk 10

# Qdos

In eerdere delen hebben we gezien dat er handig gebruik gemaakt kan worden van gespecialiseerde routines. Nu is de QL standaard al uitgerust met een groot aantal van deze routines. Zij zorgen voor de meest uiteenlopende zaken. Samen vormen ze de QDOS.

Qdos is een verzameling routines die de gebruiker afschermt van de machine.

De gebruiker hoeft niet te weten waar het begin zit van het schermgeheugen, hij hoeft niet te weten hoe hij dingen naar de microdrive moet schrijven. Er zijn een aantal routines die dat voor hem doen.

Het enige dat de gebruiker hoeft te doen is te specificeren wat er moet worden gedaan, en in welke volgorde, en tenslotte welke argumenten hiervoor nodig zijn. Qdos regelt dan de feitelijke uitvoer. Het is ook heel belangrijk dat alles wat U uitvoert gebeurt via QDOS. Als bijvoorbeeld een routine aanroept die een karakter op het scherm zet dan houdt QDOS ook bij waar een volgende karakter moet verschijnen, en hij houdt bij of het nog wel mogelijk is om een karakter op dezelfde regel past, of dat het naar de volgende regel moet worden verplaatst. Zou U zelf een karakter op het scherm aanbrengen, omdat U toevallig weet waar het schermgeheugen zit, dan is het niet ondenkbaar dat QDOS een volgend karakter er dwars overheen schrijft. Hij weet immers niet beter dan dat deze plaats nog vrij was.

Natuurlijk is dit een eenvoudig voorbeeld. Er kunnen desastreuze dingen gebeuren als U zelf op een microdrive probeert te schrijven, of zomaar in het geheugen gaat schrijven. Voor dit laatste dient U een QDOS routine aan te roepen die een stuk geheugen reserveert. Zou U dit niet doen, dan bestaat het gevaar dat een ander programma -de QL kan tenslotte meerdere programma's tegelijk laten draaien- het door U illegaal in gebruik genomen geheugen toegewezen krijgt, en uw gegevens

gaat overschrijven. Voor veel mensen is het even wennen aan een hele set voorgedefinieerde dingen, maar als je er even mee aan het werk bent, blijkt het een zeer krachtig en handig hulpmiddel. Omdat de routines vaak gebruikt worden is het van belang dat ze zo efficiënt mogelijk zijn. Dat dit inderdaad het geval is, kan U bijvoorbeeld zien door zelf eens een poging te wagen om een lijn op het beeldscherm te tekenen. Dit kunt U bijvoorbeeld doen door de juiste bits van het scherm geheugen te vullen met de gewenste kleur.

Alhoewel het trekken van een rechte lijn tamelijk eenvoudig is, is het mij nog niet gelukt om een lijn te tekenen met een snelheid die ook maar bij benadering in de buurt kwam van de QDOS line routine. Als U daarbij bedenkt dat QDOS ook nog rekening moet houden met dingen als 'OVER-1', lijnkleur, afmetingen van het gedefinieerde window etcetera, mag het duidelijk zijn dat deze routines een knap staaltje van programmeerwerk betekenen. De 68000 is standaard niet uitgerust met de zegening en van floating point rekenkracht. Daarom zijn er een aantal routines in QDOS aanwezig die het leven in deze wat vergemakkelijken. Als U twee floating point getallen wilt vermenigvuldigen, dan 'zet u ze op de arithmetic stack' (dit gebeurt op een vooraf vastgelegde wijze), waarna U de betreffende floatingpoint vermenigvuldig routine aanroept. QDOS doet dan de berekening, en geeft U bij terugkeer het antwoord op stack. U kunt het op uw beurt er dan weer van af halen, en bijvoorbeeld laten

uitprinten op het scherm. Dit laatste gebeurt, zoals we al weten, ook weer met een QDOS routine.

U ziet dat QDOS nogal een belangrijke rol vervult in de machine. U zou kunnen spreken van 'het geweten' van de machine (had ik het maar geweten...). Een goed boek over dit stuk software is van de hand van Adrian Dickens, QL Advanced User Guide. Wilt U gebruik maken van QDOS (en daar kunt U niet omheen als U op de QL in machinetaal wilt programmeren), dan is dit boek een 'must'. Er zijn ongetwijfeld een aantal andere boeken over QDOS, en allemaal hebben ze 90% dezelfde gegevens. 10% verschilt, en daarom ben je gedwongen om meerdere boeken aan te schaffen. Aangezien Dickens voornamelijk uitlegt hoe het globaal werkt, en in mindere mate alle details opsomt, kan het nuttig zijn om het Sinclair geschrift QL-Technical Guide bij de hand te hebben, maar zoals gezegd, 90% staat ook al in Dickens.

Andrew Pennell heeft ook een boekje geschreven, dat is uitgebracht door Sunshine. Het heet Sinclair QDOS Companion, en gaat meer nog dan Dickens in op globale werking. Waar Dickens vaag technisch blijft, wil Pennell nog wel eens wat voorbeelden erbij halen. De 10% verschil in dit boek is echter niet terug te vinden in de beide andere boeken. Met andere woorden: wilt U voor 150% op de hoogte komen van wat zich afspeelt in de QL, dan zou U eigenlijk 4 boeken moeten raadplegen.

Zelf hou ik het op Dickens. Dik is het overigens ook.

ARD JONKER



# OMNI READER

## OVER DE OMNI-READER en een daarvoor gemaakt programma: orfp Hieronder zijn teksten opgenomen die gelezen zijn met de OMNI-READER,

opgeslagen op RAM en ten slotte in QUILL geïmporteerd via FILES, dan weer op RAM opgeslagen en ten slotte gemerged. Het resultaat is geflatteerd doordat sommige regels meermalen zijn 'gescand' en het beste product is gebruikt.

We stuiten telkens weer op producten die voor IBM en zijn compatibelen bestemd zijn en die verkopers haast weigeren af te staan als je ze vertelt dat ze op het weeskind onder de computers aangesloten moeten worden.

Eerdere ervaringen leerden dat het wel meevalt en dat 'alleen' de software moet worden ontwikkeld.

De aankoop van de OMNI-READER vond plaats bij Funtronics. Erbij werd een los een gecopieerde gebruiksaanwijzing en een 5 1/4 inch schijf geleverd. In de doos bleken bij het uitpakken nog wat gebruiksaanwijzingen en weer een schijf te zitten. Helaas bleek de trafo voor 110 volt. Terug in de zaak werd na een onderzoek in een aantal dozen een 220 volts exemplaar gevonden. Een seriële kabel was al eerder op voorhand aangeschaft. Merkwaardig was dat de O-R niets bleek te kunnen lezen van wat ik met de OKIMATE20 op thermisch papier had afgedrukt, pas toen het met lint gedrukt nog eens aan hem werd voorgelegd reageerde hij weer. Dit is des te vreemder omdat hij met, tegen copieren, rood gekleurd papier geen enkele moeite had.

In PC business INFO was de beoordeling destijds vrij negatief. Dit acht ik niet helemaal terecht. Het overtypen van stukken is vrij vermoeiend doordat de tekst gelezen, onthouden en getypt moet worden, waarbij het oog toch telkens moet wisselen van papier naar scherm.

Dit kan de O-R je besparen. De software is uiteraard niet voor QL ogen bestemd. Wat staat erop? Op kantoor gelezen bleek de door de winkel gemaakte copie iets uitgebreider dan

die uit de doos en bevatte programma's uit de jaren '80 en '84. Daaruit mag blijken dat we hier nog wel wat op de USA, waar hij vandaan komt, achter lopen.

Erg veel zinnigs lijken de schijven niet te bevatten. Er zijn programma's om teksten direct in tekstverwerkers en spreadsheets in te voeren.

Het eerste is, indirect, dus in QUILL al gelukt. Direct ging het niet, al heb ik het niet intensief geprobeerd.

De grote vraag is echter hoe andere dan de vier standaard lettertypen moeten worden ingebracht.

Logisch zou zijn geweest in de O-R het commando EXTRA te scannen dan de betreffende letter en daarop uit de computer aangeven om welke letter het gaat. De handleiding geeft echter vaag aan dat vanuit de computer de grafische opbouw van de letters in de O-R moet worden geladen. Wellicht kan dit met een FONT-EDITOR, wellicht wordt het zelfde grafische systeem als in OKIMATE20 gebruikt: letters in een matrix en twee waarden per kolom. Als iemand het weet, hoor ik het graag.

### Voorbeelden van gelezen teksten uit de handleiding:

LAZY DOG THROUGH THE OMNI-READER INTO YOUR MICRO COMPUTER SYSTEM AND ONTO THE SCREEN WITH AS LITTLE EFFORT AS IT TAKES TO READ 'THE QUICK BROWN FOX' the quick brown fox jumps over the lazy dog through the omni-reader into your microcomputer system and onto the screen with as little effort as it takes to read 'the quick brown fox' 1234567890

Gelezen van door OKIMATE20 met lint gedrukte tekst, niet exact het lettertype waarop de OMNI-READER geprogrammeerd, 9 fouten: Ten &evol&e van de niet aflatende drane de QL van randapparatuur te voorzien stuit ik telkens weer producten die voor IBM en zijn compatibelen bestemd zijn en die ver-

kopers haast weigeren te verkopen als je ze vertelt dat ze op het weeskind onder de computers aangesloten moeten worden. Gecompiled kan het programma het scannen vlot aan, omdat met het verschuiven van de lineaal ongeveer evenveel tijd gemoeid is als QL en OR nodig hebben om een regel in te voeren.

J. Deenik,  
Gruttolaan 4,  
2261EV Leidschendam

Indien er voldoende belangstelling voor het programma ORFP (de besturing van OMNI READER in combinatie met een QL) bestaat dan kunnen wij dit plaatsen in een van de volgende nummer van Quasar of het is ook mogelijk dat dit in de Cartridge-service of op de Databank geplaatst wordt. Dus indien geïnteresseerd neem dan even contact op met J. Deenik of de redactie van Quasar.

## BIJEENKOMST Sinclair- Gebruikersgroep 't Gooi e.o.

De Sinclair gebruikersgroep voor 't Gooi en omstreken organiseren regelmatig gebruikersdagen in Buurthuis De Malbak

Blaricum Byvanck

Bij winkelcentrum de Balken

Deze gratis toegankelijke bijeenkomsten voor sinclairgebruikers worden gehouden op

zondag 18 september

zaterdag 8 oktober

zaterdag 5 november

telkens van 11.30 - 16.00 uur

zaal open 10.00 uur

Bereikbaar via A27 afslag

Blaricum Huizen

Buslijn 134

### Voor meer informatie:

Oleg Holl 02153 - 13186

Arnold Spreekmeester

02152 - 62389



# PRINTERDRIVERS

Op een warme zondag avond zat ik door mijn Quasars te bladeren en kwam op pagina 681 een vraag tegen betreffende Quill en haar printer drivers. Dit laatste schijnt voor velen onder het beginners-corps, niet negatief bedoeld, een regelrechte kwelling te zijn. Dit kan ik me erg goed voorstellen want toen ik voor het eerst wat uit mijn printer wou krijgen wat overeen stemde met mijn gedachten heeft mij dit ook enige avonden gekost.

Toegegeven ik had toen wel zeer vreemde eisen. Ik wil nu een verwoede poging wagen om deze kwellende materie iets minder kwellend te maken.

Voor de goede orde : als er iets staat in de geest van type "mdv1\_" dan bedoel ik te zeggen dat je mdv1\_ in moet tikken, zonder de quotes dus, met "<ENTER>" bedoel ik te zeggen : type de ENTER toets in. Positioneer je achterwerk op een stoel voor je QL en zet haar aan, zonder floppies of cartridges. Probeer nu het programma INSTALL\_BAS dat op de QUILL cartridge staat te runnen. Als dit is gelukt vraagt ons wonder der techniek ons iets in de geest van "Install on which mdv. ". Leuke vraag niet waar, maar wat bedoeld ie nu te zeggen ?? Aan het einde van het programma heeft ie voor ons een printer-driver gemaakt die die ergens kwijt wil, wij moeten hem nu vertellen waar ie dat kwijt kan. Wil je het op een cartridge hebben in mdv1\_ type dan "1", voor mdv2\_ moet je "2" typen. Plaats nu een cartridge in mdv2\_ en type "2" gevolgd door een "<ENTER>".

Het wonder is geschied want het programma gaat verder, het vraagt ons nu voor welke poort het gemaakt moet worden. Dit zal meestal voor ser1 of ser2 zijn, dit zijn seriele, RS232 poorten.

## Wat is RS232

RS232 is een afspraak omtrent hoe we data naar de printer zullen sturen.

Zoals je al weet worden karakters voorgesteld door combinaties van nullen en eenen en het zal je dan ook niet verwonderen dat die nullen en eenen naar de printer gestuurd worden met behulp van spannings nivo's. Een van de vele afspraken die voor een RS232 poort gelden is dat een logische 1 voorgesteld wordt door een spanning die ligt tussen de +3 en +12 volt. Een logische 0 wordt voorgesteld door een spanning die ligt tussen de -3 en -12 volt, (de spannings waarden kunnen niet geheel correct zijn dit ten gevolge van mijn geheugen dat zich af en toe gedraagt als een vergiet). Er zijn nog veel meer afspraken gemaakt voor een RS232 poort maar die willen we eigenlijk ook niet weten.

Type nu "<ENTER>" om aan te geven dat het iets wordt voor ser1 of ser2. De wonderen zijn de wereld nog niet uit want het programma gaat verder met lezen van mdv1, dit duurt meestal even, wat zou die echter lezen ??, printer-drivers. Hoogst interessant, maar wat zijn dat ???

## Printer drivers

Printer-drivers zijn, populair gezegd, vertaal programma's, zij vertalen de taal van de QL naar de taal van de printer. Nu zul je je afvragen waarom er niet 1 printer driver is, een printer is toch een printer??? Ja en nee, een printer is een printer maar net zo als met de programmeer taal BASIC waar de eene BASIC de andere BASIC niet

is, is de eene printer ook de andere printer niet. Er zijn vele overeenkomsten maar ook vele verschillen.

Deze verschillen maken het noodzakelijk dat elke printer zijn eigen printer-driver krijgt. Er is in QUILL echter maar 1 printer-driver, willen we echter dat deze eene printer-driver vele verschillende printers aan kan sturen dan moeten we haar eerst volstoppen met gegevens, data, betreffende de printer(s) die zij aan moet sturen.

Dus eigenlijk maakt het programma INSTALL\_BAS geen printer-driver maar verzameld het data voor de printer-driver die het dan weg schrijft als een printer-driver-data file.

Als QUILL nu aan het printen wordt gezet dan zal zij eerst de printer-driver-data file van cartridge ophalen voordat zij begint te printen. Deze file heet PRINTER\_DAT, zie mdv1\_.

Er is ook nog een file die alle printer-driver-data files bevat die al eens een keer zijn ingevoerd, deze heet INSTALL\_DAT. We zien nu op het scherm een menu met alle printer-driver-data files die zij al kent. Wij gaan nu een nieuwe printer-driver-data file maken die we "aap" gaan noemen. Aangezien zij nog niet in de lijst voorkomt valt zij onder de kreet OTHER, ga er met de up en down arrows op je toetsen bord naar toe. Sta je eenmaal op OTHER vertel de QL dan dat je er wat in wilt editen, dit doe je door "<F2>", = edit zie boven aan het scherm), in te typen. Als je dit gedaan hebt krijg je een nieuw scherm voor je ogen met de meest weersinwekkende kreten, loop nu niet weg; we slaan ons er wel doorheen.

Laten we de lijst van boven naar beneden doorlopen; Het eerste item is "DRIVER NAME", hiermee wordt bedoeld de naam die we deze printer-driver-data willen geven, we hadden afgesproken dat we haar "aap" zouden noemen, dus we typen "LEFT ARROW", "AAP" gevolgd door een "<ENTER>".

Tik nu eenmaal op de down arrow en je komt bij het tweede item, t.w. PORT. Hiermee moet je aangeven aan welke poort je printer is aangesloten, tik eens een paar maal op de



right of left arrow en je zult alle mogelijke mogelijkheden voorbij zien flitsen. We willen onze printer aansluiten op ser1 dus we tikken net zo vaak op de right of left arrow key tot dat ser1 er staat.

Tik op de down arrow en we gaan naar het volgende item : **BAUDRATE**. Mooi woord niet waar, doch wat zou het zijn ??? Baudrate staat voor "bits per seconde", zo als ik al vertelde worden de nullen en eenen, bits, met behulp van spannings nivo's overgezonden. Willen we nu een 1 naar de printer zenden dan moeten we het spannings nivo laten stijgen naar zeg +12 volt, we moeten dit voor (1/'zoveelste') seconde op dit spannings nivo houden. We weten nu dat als we een bit overwillen sturen we het bijbehorende spannings nivo voor (1/'zoveelste') seconde op dat nivo moeten houden. In een seconde kunnen we dus 'zoveelste' bits verzenden. Voorbeeld : stel dat we onze karakters verzenden met 9600 baud, dan is dus de BAUDRATE 9600 baud, voor elk bit geldt nu dat we haar voor 1/9600 seconde op haar bijbehorende spannings nivo moeten houden. Als elk karakter uit 8 bits bestaat dan kunnen we dus  $9600/8 = 1200$  karakters in een seconde verzenden, dit komt ongeveer overeen met  $1200/80 = 15$  regels. Om nu de snelheid van de QL in te stellen gebruiken we de left/right arrows, we fietsen hiermee over alle snelheden die zij kent. Zijn we aangekomen bij de snelheid waarme we willen verzenden dan geven we een "<ARROW DOWN>" en de QL snapt ons. Het essentieele hierbij is dat en QL en printer op de zelfde snelheid moeten werken, zenden cq ontvangen. Hoe stellen we de snelheid van de printer in ??? Dat is op een afstand een moeilijk te vertellen thema, lees de manual van de printer goed door en probeer iets te ontdekken met het woord BAUDRATE o.i.d. In de praktijk blijkt dat BAUDRATES meestal met DIP switches, dit zijn kleine schakelaartjes naast elkaar, worden ingesteld kijk dus ook goed naar plaatjes die dergelijke schakelaartjes weergeven.

Met de arrow down key gaan we naar het volgende item : **PARITY**. Met PARITY wordt aangegeven of we het aantal eenen even of oneven willen hebben, we kunnen ook kiezen voor MARK of SPACE parity, maar wat dat is weet ik ook niet dus daar heb ik het niet over. We versturen karakters als

kombinaties van 7 bits. De printer verwacht, meestal, 8 bits, we hebben dus nog 1 bit over. Door dit bit nu 1 of 0 te maken kunnen we er altijd zeker van zijn dat het aantal eenen even of oneven is. Voorbeeld : Stel we willen de volgende combinatie verzenden 0010110. Het aantal eenen is hier oneven, als we dus een even pariteit hebben moeten we er nog een 1 bij aan plakken. Zouden we oneven pariteit hebben dan moeten we er een 0 bij plakken. Je zult je nu afvragen wat het nut hiervan is, de printer kan aan de hand van de pariteit vaststellen of een karakter goed overgekomen is. Als de pariteit fout is kan de printer nog een keer om hetzelfde karakter vragen of er helemaal mee stoppen. Dit is zeer printer afhankelijk, in de praktijk echter zal bij gewoon huis, tuin en keuken gebruik het karakter altijd goed overkomen. Net zoals bij de baudrate zal ook de pariteit voor de printer en QL gelijk moeten zijn, zie ook hiervoor de printer manual. Fiets met de left right arrow keys over de diverse mogelijkheden, geef een "<DOWN ERROR>" als de gewenste pariteit zichtbaar is.

Het volgende item : **LINES/PAGE**. Deze is vrij simpel, hier moet je namelijk het aantal regels invoeren dat op het papier kan, gebruik de LEFT en RIGHT ARROWS om het goede aantal lines/page instellen. De volgende is **CHARACTERS/LINE**. Deze parameter geeft het aantal karakters dat op een regel kan, dit is meestal 80, gebruik de LEFT en RIGHT ARROW keys om het goede aantal CHARACTERS/LINE te selecteren hierna een tik op de arrow down key, we komen dan bij het volgende item :

**CONTINUOUS FORM** : dit wil zeggen gebruik je ketting formulieren --> "YES", en als je losse velletjes van de HEMA gebruikt --> "NO", gebruik de LEFT en RIGHT ARROW keys om YES of NO te selecteren.

De volgende is **END OF LINE CODE**. Als we een <ENTER> in QUILL intikken zien we dat er twee dingen gebeuren, ten eerste gaat de cursor naar het begin van de regel. Vervolgens wordt naar de volgende regel gesprongen. In ASCII codes wil dit zeggen, CARRIAGE RETURN, CR, en LINEFEED, LF. Er zijn printers die bij een CARRIAGE RETURN automatisch een LINEFEED genereren, doch de meeste doen dit niet. Speel op save en tik hier, "<ARROW RIGHT>", "C", "R", ",", "L", "F", "<ENTER>".

Hiermee geven we aan dat als zij aan het einde van de regel is gekomen zij een CARRIAGE RETURN en een LINEFEED uit moet voeren.

Na een tik op de DOWN ARROW key komen we bij : **PREAMBLE CODE**. We zullen dit behandelen samen met de **POSTAMBLE CODE**. Als we iets willen printen willen we eigenlijk het volgende doen: Initialiseer de printer met de juiste data, print de file, initialiseer de printer met een soort reset code. Het initialiseren van de printer met de juiste data, b.v. letter type, noemen we PREAMBLE. Het initialiseren met een soort reset code noemen we POSTAMBLE. Hoe je deze codes kiest ligt helemaal aan je zelf, ik heb een EPSON LX-86 en als PREAMBLE CODE gebruik ik : "ESC", "@", "ESC", "R", "NUL", wat zoveel wil zeggen als : Reset de printer en gebruik de internationale karakter set. Ik gebruik geen POSTAMBLE code. Na POSTAMBLE CODE volgt **UNDERLINE ON, UNDERLINE OFF, SUBSCRIPT ON EN SUBSCRIPT OFF**.

Hiermee kunnen we de QL vertellen hoe zij het onderstrepen op de printer aan cq uit moet zetten evenals subscript aan en uit. Voor de codes hiervoor kijk in je printer manual.

Als laatste krijgen we 10 maal een translate functie. Deze translate werkt min of meer het zelfde als de REPLACE functie van QUILL, doch wel met de restrictie dat zij slechts 1 karakter kan vertalen in 1 of meer karakters, maximaal 9 karakters een karakter mag ook "ESC" zijn. Met deze translate functie kunnen we de string "Piet" dus niet veranderen in "Klaas". Het is echter wel mogelijk om het copyright teken te veranderen in de string "Copyright". Als je deze translate functie wilt gebruiken tik dan eerst het karakter, of haar decimale of hexadecimale equivalent, in en daarna waar het in veranderd dient te worden waarbij alle karakters gescheiden moeten zijn door komma's. We hebben nu onze eigen printer-driver-file gemaakt, tik op "<F5>" om haar te installeren op mdv2\_. Ik hoop dat ik met dit verhaal menige beginner heb geholpen en wens je veel experimenteer plezier/geluk, voor mensen met opmerkingen/aanmerkingen of wat dan ook .. klim achter je QL en tik iets in voor de Quasar.

Arjan ten Hoopen  
Klaaskampen 4a 1251 KP Laren (N.H)



## QL HARDWARE UITBREIDINGEN

Zaterdag de 30ste net terug gekomen van vakantie las in mijn aanbod maandags in de QUASAR (post vast laten zetten). Ook heb ik enige reacties ontvangen. Dit artikeltje is dus even zeer snel in elkaar geflanst om nog op tijd te zijn voor de sluitingsdatum van het kopij. Hopelijk is het op tijd. Dus niet al te veel op het taalgebruik letten !!

Allereerst een kleine inleiding wat betreft enkele termen. Dat werd mij nl. gevraagd in een van de reacties.

### AD/DA converter

Zoals velen wel zullen weten kan met behulp van een temperatuurgevoelige weerstand de (inderdaad) temperatuur meten.

Al naar gelang de temperatuur laat de weerstand een bepaalde stroom door. Dit noemen wij analoog. Elke computer werkt echter digitaal. Dat wil zeggen alleen met nullen en enen. Dit zijn natuurlijk ook stroompjes maar deze zijn niet groter dan 5 V of kleiner dan 0 V. 5V geeft een 1 en 0V een nul. Beide hebben natuurlijk een bepaalde marge. Een AD converter zet dus analoge informatie om in digitale. Nu kan de weerman bijvoorbeeld de temperatuur/lichtinval en dergelijke constant op zijn computer volgen. Een DA converter doet natuurlijk het omgekeerde. Dit is bv. zeer nuttig voor het verzamelen van gegevens over temperatuur/luchtdruk/licht/onderzoeken van een stroom enz.

### Digitizer

Een digitizer zet een TV beeld om in voor de computer begrijpelijke informatie (digitale informatie dus). Hier mee kan je bijvoorbeeld een CamCorder laten afspelen op de computer die de hele film opslaat (niet mogelijk bij ons: je hebt dan voor een

filmpje (Z/W) van ca. 5 min ong. 450 Mb nodig !!!!!!!).

De bedoeling is dat je gewoon de TV aan hebt staan met de computer en scanner er naast en dat als je een beeld wil bewaren even op de knop drukt en klaar. Ik denk zelf gebruik te maken van een VIDEO plug voor de eenvoud.

Het VIDEO signaal is natuurlijk weer analoog (lees maar het QL WORLD artikel over de SPEM digitizer) en je hebt dus een AD converter nodig, een snelle wel te verstaan.

We hebben eentje nodig die in 1/25 sec 500000 beeldpunten afscaant, dus 25000000 per seconde. Dit vraagt om een conversie tijd van 40 ns, maar of dit snel genoeg is weet ik nog niet, want de computer moet immers steeds de opdracht geven en teruglezen. En als je een hogere resolutie wilt een nog snellere...

**Gevraagd:** datasheet met preciese beschrijving van het VIDEO signaal (ik weet ongeveer hoe het in mekaar zit maar een boekje over TV techniek kost f 42,- terwijl ik daarvan maar 5 bladzijden nodig heb) opsturen a.u.b.

Overigens er komt natuurlijk ook software bij de digitizer, die bruikbaar is op het gewone scherm alleen moet dan wel scrollen, het hele beeld past er niet op.

Een Z/W beeld vraagt ca. 62 Kb, dus 32 Kb in schermgeheugen, 30 Kb in vrije RAM, het wil dus wel redelijk op een 128 Kb QL maar niet als er met kleur wordt gewerkt.

Stel slechts vier verschillende kleuren. Daarvoor zijn twee bits nodig, dus 124 Kb voor een TV beeld (625 x 800).

### Tweede scherm

Ook werd door iemand getwijfeld over het nut van een tweede scherm voor de QL.

Stel we hebben een TV beeld gedigitaliseerd. Laten zeggen dat het lukt (Z/W wel te verstaan) dat vraagt om een resolutie van 625(ver) x 800.

Zo'n resolutie kan de QL niet aan, daarvoor is een tweede scherm nodig. Ook is een tweede scherm natuurlijk zeer handig bij CAD en andere tekenprogramma's. Alleen je moet ze zelf schrijven.

Bovendien kan je ervoor zorgen dat er op het 2e scherm meer kleuren kunnen deze zeer snel is e.d.

### Scanner

Deze doet in feite doet deze het zelfde als een digitizer, maar nu is de input geeft video/TV/CamCorder beeld maar bv. een foto. De snelheid verschilt natuurlijk per te scannen afbeelding. En hoeft niet zo snel te zijn, een foto beweegt immers niet meer...

### Harddisk/floppy disk interface

De harddisc of floppydisc interface is een van de moeilijkste projecten, kwa programmeren (hier begin ik dan ook later mee). Dus ik hoop dat alles lukt. Overigens wil ik wel even wat kwijt over microdrives/floppy/harddisc. De microdrives zijn traag en hebben zeer weinig opslagcapaciteit. Nu zal voor de meesten de traagheid wel mee vallen, machine taal programma's laden snel genoeg en voor SuperBasic zijn er goedkope programmaatjes die de load en save tijden dramatisch terugbrengen. Voor de beperkte opslagcapaciteit is geen remedie.



Backups zijn zeker bij cartridges absoluut noodzakelijk. Ik heb nu zelf 52 cartridges (en nog eens 5 onderweg) en ik doe met pijn en moeite afstand van de cartridge waarop dit artikel staat. Ik heb al herhaaldelijk data weggepoetst omdat ik de cartridge beslist nodig had. Later heb ik daar vaak spijt van, weer een flinke hoop typewerk te doen... Dus wat heb ik nodig? Floppydrives, want harddiscs zijn te duur. Ja of nee, even rekenen.

#### Floppydrives

De interface komt op ong. 175,- nu moet je nog drives 2 x 3.5 inch ong. (720 Kb geformatteerd) ca. 480,-. Totaal 655 gld en dan heb je nog een voeding en kast voor de drives nodig.

Dus samen ong. 755,-

#### Harddisc

De interface komt schat ik op ongeveer het zelfde ook 175,-

En nu een drive van laten we zeggen 20 Mb met een normale toegangstijd. De interface zal (dat is de bedoeling) de SCSI standaard ondersteunen dus kan je kiezen uit honderden drives. Even kijken in PCM 800,- maar dan wel met een controller, dus zonder controller nog goedkoper laten we zeggen 525,- (een wel hele goedkope controller...). Hier komt nog een voeding bij plus kastje voor de drive. Dus met z'n allen f 800,- (oh ja toegangstijd gemid. 65 ms). Uitkomst Het verschil tussen harddisc en floppydisc valt dus wel mee. Natuurlijk hebben beide voor- en nadelen. Zo heeft harddisc maar een beperkte opslagcapaciteit, bij floppydrives koop je er gewoon diskettes bij. Maar een harddisc is betrouwbaarder en veel sneller boven voor 20 Mb heb je bijna 30 diskettes nodig en voor een beetje kwaliteit kom je op f 150,-.

Floppy's kan je natuurlijk makkelijk meenemen, maar ik heb bv. nog cartridges zat daarvoor. De bedoeling is overigens dat het een gecombineerde floppy/ harddisc interface wordt...

#### MIDI

MIDI staat voor Musical Instrument Digital Interface. Het is een soort seriële interface (data staat achter elkaar) die met 32 Kb/s data over de lijn stuurt. Deze interface wordt zoals de naam al zegt gebruikt voor het versturen van informatie van en naar

keyboard/mixer/drum machines/ sequencers enz.

#### Andere projecten

Ook kan je nog denken aan een robot-besturing / productie-besturing/ batterij klok/plotter enz...(wat zeg ik hier wel niet allemaal...)

#### Buffer & RAM

Hier mee zal ik beginnen. De bedoeling is dat er een kastje komt met slots, net zoiets als bij de IBM PC's (wat zet ik er nu weer neer...). Een kast zal ruimte bieden aan een voeding extra RAMEPROM (zie verderop ook nog)/12 slots voor printkaarten. De slots zijn natuurlijk gebufferd, zie artikel van Spanjer. De RAM is ook zeer simpel te maken en ook eigenlijk wel nodig voor serieuze toepassingen (kleuren digitizer), deze gebruikt zoals bekend niet een speciaal I/O slot. De extra EPROM sockets zijn voor als nog gewoon de verplaatsing van de ROM slot achterop de QL en directe aansluiting voor de zgn.

#### I/O ROM.

Misschien, als iemand mij kan vertellen hoe en wat ook voor andere EPROMs. QL Connexions Naar ik heb gelezen is er geen belangstelling voor de QL Connexions manier, makkelijker voor me...

Maar meer voor een duidelijke print/software/beschrijving.

Voor dit alles moet ik natuurlijk wel een bepaalde investering doen en helemaal als het niet lukt. Natuurlijk kan ik vele dingen combineren, zoals de parallele&keyboard interface. Daarom zal ik bij sommige projectjes (parallele/keyboard...) die ik zelf niet gebruik of echt grote ingewikkelde een kleine betaling verlangen voor de software. Ja nou krijgen we het, er ook nog grof geld aan willen verdienen, denkt iedereen al. Natuurlijk niet, de meesten zullen wel niet beschikken over een EPROM programmer (ik ook nog niet) en eigenlijk moe-

ten daar de devicedrivers in staan, dus ik zal wel voor enkele dat moeten doen. Ik denk zelf aan een bedrag van f 5,- voor de kleine die ik zelf niet gebruik en f 10,- voor de grote ingewikkelde tijdrovende projecten. Dit lijkt mij allemaal zeer redelijk.

#### Seriële - parallell converter

zit er niet in, wel hebben er in Elektoer verscheidene gestaan...

Geen garantie Dit hele verhaaltje staat er nu wel zo mooi maar, ik geef er geen garantie op dat het er ook komt.

Dus...

Peter Venema

Westereems 1

9642 KP Veendam

bij bellen gaat de stekker eruit...

#### Te koop aangeboden:

SINCLAIR QL 128K (met een uitstekende koeling tegen oververhitting van computer + voeding)

JOYSTICK interface + joystick QUICKSHOT II

Centronics printer interface

Het Sinclair's QL handboek

Het Sinclair's QL leren programmeren in basic

QL user Guide

Het Quasar tijdschrift van 1 jaar

15 tijdschriften QL world

8 tijdschriften Sinclair Gids

12 lege cartridges

21 cartridges boordevol programma's en files

cartridge bewaarbox voor 20 cartridges.

Voor dit (liefst in een koop) vraag ik ± 16.000 Bfr.

Bart Devroey

Boterbergstraat 61

B - 3261 WAANRODE (België)



# TANDATA QCONNECT

Robert Lokerse

Aanleiding van dit artikel is een aantal plezierige ervaringen die ik heb opgedaan nadat ik een tweede QL via het netwerk met de eerste verbonden had.

Naar alle waarschijnlijkheid zijn er zo onderhand wel een hoop QL gebruikers die meer dan n machine hebben staan en net als ik deze machines via het netwerk willen koppelen.

Een voorwaarde voor het 'plezierige' is de aanwezigheid van Tony Tebby's TOOLKIT II (in eprom !!!) op beide machines.

Mijn configuratie is als volgt:-

1 QL (JS) & kleurenmonitor met de TRUMP CARD & dual 3.5 inch NEC drives en ICE eprom. (in de TRUMP CARD zit oa. TOOLKIT II versie 2.11)

1 QL (JM) & monochroom monitor met een 256 Kb. PCML disk-IF & TOOLKIT II eprom (versie 2.10) Tevens is mijn TANDATA modem aan deze machine verbonden.

Beide machines zijn verbonden door het bijgeleverde kabeltje voor het netwerk. Het maakt niet uit welke net-aansluiting je gebruikt.

In de rest van het verhaal zal ik de QL met TRUMP CARD aanduiden als de 'grote QL' en de PCML uitvoering als de 'kleine QL'.

Het belang van de toolkits in eprom ligt in het feit dat de 'server' die tot je beschikking staat niet werkt als je een illegale, van eprom gekopieerde toolkit gebruikt. Ik weet niet hoe de gedragingen zijn van een originele ramversie van TOOLKIT II.

De reden om het modem aan de 'kleine QL' te verbinden is het onvoorspelbare gedrag van de Qconnect software in combinatie met de TRUMP CARD. Ook na een RES\_128 is een en ander niet helemaal zoals de software verwacht. De floppy drives zijn 9 van de 10 keer niet te gebruiken. Hoe dit kan is door mij niet te verklaren. Ik was tot voor kort ook nog niet in het bezit van de prima terminal software

van Andre van de Wijdeven, wat speciaal voor het TANDATA modem werd ontwikkeld. Het nadeel van deze software is het niet ondersteunen van 'viditel-achtige' databanken.

Na het omzetten van het modem van de 'grote QL' naar de 'kleine', werkt alles zoals door TANDATA bedoeld. Zoals misschien bekend is de Qconnect-code niet relocatable en moet dus op een vaste plaats in het geheugen zitten. De geheugenruimte wordt met RESPR(x) vastgelegd en de hulp-programma's bij Qconnect zorgen ervoor dat dit op de juiste plaats gebeurt. Een QL met meer dan 128 Kb. wordt software-matig naar 128 Kb. 'gereset'. Voor QL-gebruikers zonder floppy's maar wel met geheugenuitbreiding, een onplezierige zaak. Files 'spoolen' van en naar microdrive is een slijtageslag voor de cartridges en kost een hoop tijd.

Na wat experimenteren heb ik, met behulp van de toolkits en een tweede QL, een goed werkende oplossing voor de tekortkomingen gevonden. Hierna volgen enkele kleine basic programma's die bedoeld zijn voor het netwerk. Op een enkele machine werken met deze programma's levert geen voordeel op. Het is namelijk een probleem om files uit de ramdisk veilig te stellen daar men niet op een normale manier, d.w.z. zonder een reset te veroorzaken, Qconnect kan verlaten!

Het eerste programma is een BOOT file voor de 'grote QL'. Hierin worden een aantal aanwijzingen gegeven en in te stellen waarden gezet.

De toolkit wordt genitaliseerd, het netwerknummer wordt gezet op 1, de server wordt gestart en de route naar de microdrives wordt omgeleid naar

de ramdrives op de andere machine.

```
2 REMark DELETE
  flp1_BOOT:SAVE flp1_BOOT
3 REMark DELETE
  mdv1_BOOT:SAVE mdv1_BOOT
4 IF VER$='JM':PRINT#0\, 'Dit
  BOOT-file is bedoeld voor de NET
  1 machine !':STOP
5 TK2_EXT:NET 1:
  FSERVE:PRT_USE SER, SER:
  NFS_USE
  mdv,N2_RAM1,N2_RAM2_
6 FOR x=1,2:
  WINDOW#x,512,204,0,0:
  PAPER#x,0:INK#x,7:CLS#x:
  END FOR x
7 PRINT\ TO 25;-- Q C O N N E C
  T via network --\
8 PRINT\ Time & Date
  today:--\,DATE$
9 PRINT\ 'The Network Server is
  running.'
10 PRINT\ 'Serieel Print Buffer is
  initialized.'
11 PRINT\ 'Routing MDV to
  N2_RAMx is set.', '(Reference to
  MDVx will affect'
12 PRINT\ ,,,,, ' N2_MDVx ! (i.e.
  Within ICE))'
13 PRINT\ 'Net number of this
  machine is 1.'
14 PRINT\ 'Type on the Net 2
  machine:-'
15 PRINT\ ,LRUN N1_FLP1_Qcon1
  <ENTER>
16 PRINT#0\, 'ANY KEY WILL
  CLEAR THIS SCREEN!'
17 PAUSE:CLS#0:CLS#1:CLS#2:
  ICE
```

Ik ga er van uit dat de tijd op de 'grote QL' goed loopt. Deze tijd zal door de 'kleine QL' opgehaald worden. (Hiero-ver straks meer.) Het BOOT file wat bij Qconnect geleverd wordt gebruik ik niet meer. De files B2, B3, en B4 heb je ook niet meer nodig. Inplaats hiervan gebruik ik nu Qcon1 en Qcon2. Gebruikers met een QL en een JS rom kunnen het met alleen een BOOT file af. Dit wordt dan een gedeeltelijke combinatie van Qcon1 en Qcon2. Als het BOOT file ingeladen is wordt op de 'kleine QL', direct



na het indrukken van F1 of F2, LRUN N1\_FLP1\_Qcon1 ingetyped. Over het netwerk zal nu het file Qcon1 van floppy geladen worden in de 'kleine QL'. De tijd wordt van de 'grote QL' gelezen door procedure NET\_TIME, er wordt een driver voor dynamische ramdisks geladen, de server wordt gestart en vervolgens wordt Qcon2 over het netwerk van floppy gehaald. De opzet van Qcon1 en Qcon2 is nodig omdat de server niet direct genitiseerd kan worden doordat er een 'bug' aanwezig is in JM rom's. Op een JM werken de in de basictable 'ingelinkte' routines alleen goed na een NEW, LOAD of LRUN.

```

2 REMark DELETE
  mdv1_Qcon1:SAVE mdv1_Qcon1
3 REMark DELETE
  flp1_Qcon1:SAVE flp1_Qcon1
4 IF VER$=JS:PRINT 'Dit pro-
  gramma werkt niet met de
  TRUMP CARD!':STOP
5 FOR x=1,2::
  WINDOW#x,512,204,0,0:
  PAPER#x,0:INK#x,7:CLS#x:
  END FOR x
6 NET 2:NET_TIME:LRESPR
  n1_flp1_dynaram_bin:Fserve:L
  RUN N1_FLP1_Qcon2
7 :
8 DEFine PROCedure NET_TIME
9 CH%=FOP_OVER
  ('N1_RAM1_**TIME**')
10 PRINT#CH%,DATE$:CLOSE#CH
  %
11 CH%=FOP_IN
  ('N1_RAM1_**TIME**')
12 ADATE(-DATE):ADATE FUPDT
  (#CH%) 13 CLOSE#CH%:DELETE
  'N1_RAM1_**TIME**'
14 END DEFine NET_TIME

```

De regels 4 van het BOOT- en beide Qconx programma's heb ik er eigenlijk alleen voor mijzelf ingezet. Heb je twee QL's met een gelijk rom dan gaat dit niet op natuurlijk. Het is een heel primitive manier van controleren. In Qcon2 wordt gebruik gemaakt van de functie ALCHP (Allocate Common Heap). In tegestelling met de superbasic functie RESPR, welke geheugen reserveert van 'boven' naar 'beneden', zal ALCHP geheugen reserveren van 'beneden' naar 'boven'. We kunnen nu dus een stuk geheugen afschermen voor QDOS en toch niet-relocatable code 'laag' in het geheugen zetten. Wat je moet weten is het startadres van de code en de lengte plus de dataspace van de code. In het geval van Qconnect weet ik het startadres (179792) en de

lengte (82169 bytes). Als ik dit bij elkaar optel krijg ik 261961. Dit is slechts 183 bytes onder 'ramtop' (262144) van een standaard QL. Ik reserveer nu met ALCHP zoveel geheugen dat het startadres onder adres 179792 ligt en het laatste adres van de gereserveerde ruimte gelijk is aan ramtop van een standaard QL, 262144. Deze waarde, a op regel 6 in Qcon2, is proefondervindelijk vastgesteld.

Op de QL met de TRUMP CARD gaat dit niet op omdat er niet 'laag' genoeg geheugen gereserveerd kan worden. Dit komt waarschijnlijk doordat er allerlei truks toegepast zijn om alle extra's te verkrijgen. Als de TRUMP CARD 'gereset' naar 128 Kb. met het commando RES\_128, dan gaat het wel. Dit is nuttig als het TANDATA modem is verbonden met een QL met TRUMP CARD en niet aan de andere, in het net opgenomen QL.

Qconnect wordt nu over het netwerk ingeladen op adres 179792, de klok wordt ingeladen (originele naam is clk), en het startadres van Qconnect wordt aangeroepen. Binnen Qconnect heb ik nu de beschikking over de microdrives, de floppy's en de ramdrives. De laatste twee op beide machines!!! Vanwege het NFS\_USE uitgevoerd in de BOOT van de 'grote QL' kan ik vanuit ICE rechtstreeks in de ramdrives van de 'kleine QL' lezen en schrijven als ik het MDV ikoon 'klik'. Het enige schoonheidsfoutje is, dat de klok niet verdwijnt als er is 'ingelodged'. Ik denk dat de procedure die dit verzorgd in Qconnect de klokcode niet kan vinden. De klokcode zit natuurlijk niet 'heel laag' in het geheugen. Je kan het opstarten van de klok natuurlijk ook achterwege laten, 'who needs the time when computing?'

```

2 REMark DELETE
  mdv1_Qcon2:SAVE mdv1_Qcon2
3 REMark DELETE flp1_Qcon2:SAVE
  flp1_Qcon2
4 IF VER$=JS:PRINT'Dit programma
  werkt niet met de TRUMP
  CARD!':STOP
5 ALTKEY'p',' ','n41N1_FLP1_',",
  CHR$(236),'37phonebook',",
  CHR$(236)
6 CLCHP:CLOSE:CLEAR:
  DEL_DEFB:a=85164:b=179792:
  c=ALCHP(a)
7 PRINT#2\,'Date & Time
  today:-\,,DATE$
8 PRINT#2\,'Qcon
  :!b,262144\,'Alchp :!c,
  (a+c)\,'Ram1/8: Dynamisch.'

```

```

9 PRINT#2\,'Free
  :!FREE_MEM!bts.',FREE_MEM/10
  24!'Kb.'
10PRINT#2\,':UNDER#2,1:PRINT#2
  , 'Type at the name-prompt: ALT &
  p '
11NET 2:CSIZE#0,2,0:INK#0,7:
  CLS#0:AT#0,0,3
12PRINT#0,'Loading Qconnect -
  Version 2.2 ..."
13COPY_O n1_flp1_phonebook TO
  ram1_phonebook
14BYTES n1_flp1_Qconnect,
  179792:CLOSE#0
15CLOSE#1:CLS#2:EX
  n1_flp1_Qclock:CALL 179792

```

Heb je alleen software voor statische ramdisks, formateer deze dan vanuit Qcon2 tussen het reserveren van het geheugen en het uitvoeren van de CALL. (b.v. direct na regel 6) De procedure NET\_TIME leest de tijd van de 'grote QL'. Met behulp van toolkit functies wordt er op de andere machine een file geopend, de tijd wordt hierin geschreven (dit mag ook achterwege blijven want het heeft verder geen enkele functie) en het kanaal wordt gesloten. Wederom wordt het nu bestaande file geopend. De klok van de 'kleine QL', waar dit programma op loopt, wordt op de begindatum gezet.

De aanmaaktijd van het nu geopende file wordt gelezen. Dit is de tijd van de 'grote QL'. Deze tijd wordt gebruikt om de klok van de 'lezende' machine te zetten. Het file wordt gesloten en opgeruimd. In deze procedure wordt een ramdisk gebruikt om tijdsverlies te voorkomen, maar het mag ook een floppy of microdrive zijn.

```

2 REMark DELETE mdv1_TIME:SAVE
  mdv1_TIME
3 REMark DELETE flp1_TIME:SAVE
  flp1_TIME
4 NET_TIME
5 :
6 DEFine PROCedure NET_TIME
7 CH%=FOP_OVER
  ('N1_RAM1_**TIME**')
8 PRINT#CH%,DATE$:CLOSE#CH%
9 CH%=FOP_IN
  ('N1_RAM1_**TIME**')
10ADATE(-DATE):ADATE FUPDT
  (#CH%)
11CLOSE#CH%:DELETE
  'N1_RAM1_**TIME**'
12PRINT\,'Date & Time
  today:-\,,DATE$
13END DEFine NET_TIME

```

Als laatste nog een programmavoorbeeld om berichten over het net te zenden.



Mocht je in het bezit zijn van Q-LIBERATOR, versie 3, dan kun je hier misschien een basic-extensie van maken. Ik heb dit nog niet geprobeerd.

```

2 REMark DELETE
  flp1_NET_MESS:SAVE
  flp1_NET_MESS
3 ALTKEY'm',"CLS#0:input#0, 'Uw
  naam? > '&sender$:print#0,'Max 61
  char. '?'&fill$('
  ',59)&'?:INPUT#0,'Uw bericht?
  >';mess$: MESSAGE(mess$),
  (sender$):
  CLS#0:PRINT#0,'Bericht
  verzonden.'"
4 PRINT#0\,,,'Bericht verzenden;
  ALT & m'
5 :
6 DEFine PROCedure message
  (mess$,sender$)
7 LOCAl nt$,channel%,L%,
  tab%,y%,x
8 IF LEN(mess$)>61:mess$= mess$
  (1 TO 61)
9 IF NOT LEN(sender$): sender$=
  'Station '&NETNR&'
10 IF NETNR=1:nt$='N2':ELSE
  nt$='N1' 11 channel%=FOPEN
  (nt$&'_CON_512x58a0x0_128')
12 BORDER#channel%,4,255:PAPER
  #channel%,4
13 INK#channel%,0:CLS#channel%:L
  %=LEN(mess%)
14 PRINT#channel%, ' Message
  from:-' '&sender$
15 AT#channel%,0,61:PRINT#channel
  %,'Date & Time today:-'
16 AT#channel%,1,61:PRINT#channel
  %,DATES
17 CURSOR#channel%,6,40:
  PRINT#channel%,*** channel
  closed ***
18 OVER#channel%,1:tab%=(
  (496-L%*8)/2):FOR x=0,7
19 y%=x DIV 7:INK#channel%,x:
  CURSOR#channel%,(tab%+y%),
  (18+y%)
20 CSIZE#channel%,1,1:
  PRINT#channel%,mess$
21 BEEP 1000,10000:
  END FOR x:CLOSE#channel%
22 END DEFine message
23 :
24 DEFine FuNction NETNR: RETurn
  PEEK(163895):
  END DEFine NETNR

```

## FOUTJES in listing Printer\_dat vorig nummer

In het programma van Dick Nieuwensteeg over het gebruik van Printer\_dat in BASIC zijn helaas wat foutjes geslopen doordat per ongeluk de voorlaatste versie geplaatst is, wanneer u de volgende wijzigingen aanbrengt zal het echter wel goed werken.

```

regel 140 PRINTER_INIT moet worden INIT_PRINTER
regel 340 BAUDRATE-CODE(PRINTER$(18))*256+CODE(P BAUD RATE
  moet worden
  BAUDRATE=CODE(PRINTER$(18))*256+CODE(PRINTER$(19)):
  REMARK BAUDRATE
regel 380 DEV$=PRINTER$(18 TO18+CODE(PRINTER$(17)))
  moet worden
  DEV$=PRINTER$(18 TO17+CODE(PRINTER$(17)))
regel 720 IF LENGTH>0 THEN moet worden IF LENGTH=0 THEN
regel 760 LINE$LINE$&printer(POS+J) moet worden
  LINE$LINE$&printer$(POS+J)

```

Zo dit waren ze Succes

## EUROPESE QL BIJEENKOMST IN BELGIË

**Op 23 Oktober 1988 is er een internationale QL  
bijeenkomst in**

**Hotel De Borggraaf in Hasselt BELGIË**

**Met deelnemers uit Duitsland, Engeland, Frankrijk, België,  
Nederland, Scandinavië en misschien nog andere landen.**

De bedoeling hiervan is wat internationale support te geven aan het QL gebeuren, ook zijn er plannen voor het opzetten van een internationale QL-Databank en er staan diverse lezingen en demonstraties op het programma.

Voor meer informatie:  
QL-gebruikersgroep LIMBURG  
M. Beckers  
Houterstraat 100  
3762 Gellick BELGIË

LEO MOLL vroeg zich af of het geen goed idee is om te kijken of het mogelijk is dat diverse QL gebruikers met

elkaar mee te rijden naar de Internationale gebruikersdag in BELGIË en hij stelt zich beschikbaar als coördinator voor de regio ROTTERDAM/DELFT en omstreken. Dus ga je naar de QL DAG in BELGIË en heb je wel of geen auto meldt je even bij LEO MOLL.

Tel. 010-4263452  
Na 18.00 uur

Mensen die als coördinator voor een andere Regio willen optreden laat dat even weten aan de redactie van QUASAR zodat wij dat in het volgende nummer kunnen publiceren.